

Copyright
by
William Alfred Bamford Jr.
2004

The Dissertation Committee for William Alfred Bamford Jr.
certifies that this is the approved version of the following dissertation:

Navigation and Control of Large Satellite Formations

Committee:

E. Glenn Lightsey, Supervisor

Robert H. Bishop

Wallace T. Fowler

Cesar Ocampo

Russell Carpenter

Navigation and Control of Large Satellite Formations

by

William Alfred Bamford Jr., B.S., M.S.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2004

Acknowledgments

First and foremost, this research was partially funded by the NASA Cooperative Agreement No: NCC5-732.

To my advisor: Dr. Glenn Lightsey: Thank you for sticking with me, even though it seemed as if I would never go away. You kept pulling for me long after most would have stopped, and I appreciate that.

To the members of my committee: Thank you for the time you spent reading, re-reading, and in some cases, re-re-reading my dissertation.

To Dr. Takuji Ebinuma: After all of your help, this dissertation is almost as much yours as it is mine. Thanks for always being around to answer my questions.

To Larry Jackson, Rich Burns, Dr. Mike Moreau, Dr. Dave Gaylor, and the guys at Emergent Space Technologies: Thanks for the remote support, and the hours on the telephone trying to make the Spirent simulator work correctly.

To the research group: Greg Holt, Jacob Williams, and Shaun Stewart: Thanks for the tips and the white board art (though I almost never understood it). Also, thanks for always being there when I needed a favor.

To my parents and in-laws: Thanks for the constant support, even from half a continent away.

To my children: Ashley, Tyler and Zachary: You always understood when I had to miss your soccer practices or baseball games. You tolerated seeing me for one hour a day, and that was usually on the drive to Mommy's work. The nights that I was home, you usually saw more of the back of my head than the front (as I worked on the computer). Yet, through it all, you were always there with a smile, a hug, and a kiss on the cheek. You three provided the motivation to keep going when things got hard. I thank you for being understanding, and still loving your Daddy through it all.

Finally, to my wife Katie: It took seven long years, but it is finally over. You bore each sacrifice alongside of me, and you never complained, no matter how tough things got. (And things couldn't get much worse than University Housing). You took care of the kids when I was gone, and took care of me when I was home. You spent seven years pulling the weight for the both of us at home, while I chased my dreams in a classroom. I couldn't have accomplished any of this without you. Thank you, and I love you.

Navigation and Control of Large Satellite Formations

Publication No. _____

William Alfred Bamford Jr., Ph.D.
The University of Texas at Austin, 2004

Supervisor: E. Glenn Lightsey

In recent years, there has been substantial interest in autonomous satellite formations, driven by the new technologies that enable smaller and cheaper spacecraft. Formation flying allows for mission designs, such as stereoscopic imaging, that are impractical or impossible for a single satellite. Much of the current work focuses upon small formations, which can be defined as four or less satellites in a relatively tight grouping. Next generation formations may be composed of more satellites spanning greater spatial distances. The large formation problem becomes more difficult for several reasons, including an increased amount of communication required between the satellites, and orbit perturbations, which become more important as the formation size grows. The purpose of this dissertation is to examine formation flying for large formations, and determine whether or not generalizations can be made linking the large and small formation regimes.

In order to model formations with many satellites, a simulation environment was constructed in which different observers, controllers, and formation architectures can be modelled. This dissertation focuses on a decentralized control scheme, but the software is general enough to accommodate a variety of control architectures. Validation of the large formation models is accomplished by initially modelling only a pair of satellites and comparing the results against those found in the literature.

As a demonstration of the theoretical results, a real-time, closed-loop, hardware-in-the-loop simulation was constructed using GPS receivers as the measurement source. A large constellation, real-time simulation system was developed that utilized the Internet to connect simulation equipment from research centers in different locations.

Table of Contents

Acknowledgments	iv
Abstract	vi
List of Tables	xiii
List of Figures	xiv
Chapter 1. Introduction	1
1.1 Background	1
1.1.1 TECHSAT-21	2
1.1.2 A-TRAIN	3
1.2 Previous Work	3
1.3 Research Contributions	5
1.3.1 Small Formation to Large Formation Generalization . .	5
1.3.2 Extended Kalman Filter Design	6
1.3.3 Demonstration of a Generalized Kalman Filter	6
1.3.4 Construction of the Extended Formation Flying Testbed	7
1.3.5 GPS Specific Instance of the EFFTb	7
1.4 Overview	7
Chapter 2. Reference Systems	9
2.1 Time Systems	9
2.1.1 Solar and Sidereal Times	9
2.1.2 Atomic time	10
2.1.3 Coordinated Universal Time	10
2.1.4 GPS Time	10
2.1.5 GPS Roll Over	11
2.2 Reference Frames	11

2.2.1	Earth Centered Inertial Reference Frame (ECI)	11
2.2.2	Earth Centered Earth Fixed Reference Frame (ECEF)	12
2.2.3	Spacecraft Centered Frame	12
Chapter 3.	GPS Models and Measurements	14
3.1	GPS Observables	14
3.1.1	Pseudorange Measurement	14
3.1.1.1	User Range Error	16
3.1.1.2	Ionospheric Delay	16
3.1.2	Carrier Phase Measurement	17
3.1.3	Doppler Measurement	19
3.2	Double-Differenced Observables	19
3.2.1	Double-Differenced Carrier Phase	21
3.2.2	Double-Differenced Doppler	22
3.2.3	GPS Satellite Propagation	22
Chapter 4.	Kalman Filter Design	25
4.1	General Formation Description	25
4.2	Filter State	26
4.2.1	Absolute And Relative States	27
4.2.2	Receiver Clock and Frequency Offset State	28
4.2.3	Double-Differenced Ambiguity State	28
4.2.4	GPS Clock Offset	29
4.3	Kalman Filter	29
4.3.1	State Vector Construction	30
4.3.2	Propagation	30
4.3.3	Initialization	32
4.3.4	Measurement Update	32
4.4	Dynamic Model and State Transition Matrix	33
4.4.1	Dynamic Model	33
4.4.2	State Transition Matrix	35
4.4.2.1	Absolute State Acceleration Partial Derivatives Φ_{Abs}	35

4.4.2.2	Relative State Acceleration Partial Derivatives $\Phi_{\mathbf{Rel}}$	36
4.4.2.3	Receiver Clock States Partial Derivatives $\Phi_{\mathbf{b}}$	36
4.4.2.4	Double-Differenced Ambiguity Partial Derivatives $\Phi_{\Delta\mathbf{N}}$	36
4.4.2.5	GPS Clock Error Model $\Phi_{\mathbf{B}}$	37
4.5	Process Noise Matrix	37
4.5.1	Absolute and Relative State Process Noise	38
4.5.2	GPS Receiver Clock Process Noise	39
4.5.3	Double-Differenced Ambiguities Process Noise	39
4.5.4	GPS Satellite Clock Bias	39
4.6	GPS Measurement Equations	39
4.6.1	Undifferenced Measurement Equations	40
4.6.2	Double-Differenced Measurement Equations	40
4.7	Measurement Noise Covariance	41
4.7.1	Pseudorange Measurement	42
4.7.2	Doppler Measurement	42
4.7.3	Double-Differenced Carrier Phase Measurement	42
Chapter 5.	Control System	45
5.1	Controller Development	45
5.2	Controller Implementation	47
5.3	Offline Controller Tuning	48
5.4	Formation Control Architecture	51
Chapter 6.	Extended Formation Flying Testbed	57
6.1	Hardware Description	57
6.1.1	GPS Signal Simulator	57
6.1.2	Orion GPS Receivers	59
6.1.3	Master Control Computer	60
6.1.4	Remote Control Computer	61
6.1.5	Hub Computer	62
6.1.6	Node Computer	62
6.2	Software Description	63

Chapter 7. Results	65
7.1 Offline Filter Validation	65
7.1.1 Filter Performance at 1 Hz	67
7.1.2 Filter Performance at 5 Seconds	74
7.1.3 Cycle Slips	79
7.1.4 Best Case Filter Performance	80
7.1.5 Receiver Clock Model Validation	86
7.2 Extended Formation Flying Testbed Results	88
7.2.1 Two Satellite Local Simulation	88
7.2.2 Multi-Satellite Formation	92
Chapter 8. Conclusion	99
8.1 Summary of Results	99
8.2 Future Work	102
8.2.1 Ionosphere Study	102
8.2.2 Multi-Path Study	103
8.2.3 Attitude Modelling and Control	103
8.2.4 Formation Control	103
8.2.5 EFFTb Extension	104
Appendices	105
Appendix A. Dynamic Equation Partial Derivatives	106
A.1 Dynamic Equation Partial Derivatives	106
Appendix B. Measurement Equation Partial Derivatives	110
B.1 Pseudorange Partial Derivatives	110
B.2 Doppler Partial Derivatives	111
B.3 Double-Differenced Carrier Phase Partial Derivatives	112
B.4 Double-Differenced Doppler Partial Derivatives	114

Appendix C. Filter Verification Procedures	118
C.1 Verification of the Dynamic Partial Derivative Matrix	118
C.2 Verification of the Measurement Partial Derivative Matrix . . .	120
C.3 Verification of Measurement Noise Settings	122
C.4 Covariance Update Verification	123
Bibliography	125
Vita	129

List of Tables

1.1	Actual and Proposed Formation Flying Missions.[22]	2
2.1	J2000 Reference System Definition	12
2.2	ECEF Reference System Definition	12
2.3	UVW Reference System Definition	13
3.1	Almanac Information	23
5.1	Controller Design Parameters	49
5.2	Initial Node Absolute State Offset From The Desired Trajectory	54
7.1	Simulator Environment Settings.	66
7.2	True Initial Conditions: November 6 2001 00:00:00 GPS.	66
7.3	Error Covariance Initialization	67
7.4	Absolute State Error in $\hat{u}, \hat{v}, \hat{w}$ frame (mean \pm std)	68
7.5	Process Noise Settings: 1Hz Update	73
7.6	Relative State Error: 1 Sec. Update Rate (mean \pm std)	73
7.7	Process Noise Settings: 0.2 Hz Update	75
7.8	Relative State Error: 5 Sec. Update Rate (mean \pm std)	78
7.9	Relative State Error: Best Case Filter Runs (mean \pm std)	86
7.10	Relative State Error: EFFTb 2 Satellite Simulation (mean \pm std)	89
7.11	Relative State Error: EFFTb 5 Satellite Simulation (mean \pm std)	95
7.12	Control Effort in the $\hat{u}, \hat{v}, \hat{w}$ frame	98
C.1	Dynamic Partial Differential Equation Validation	119
C.2	Dynamic Partial Differential Equation Validation Cont.	120
C.3	Measurement Partial Differential Equation Validation	121
C.4	Measurement Noise Verification.	123

List of Figures

3.1	Double-Difference.	20
4.1	Problem Setup.	26
5.1	Tracking Error With Too Much Control Penalty.	50
5.2	Tracking Error With Too Little Control Penalty.	51
5.3	Tracking Error With Appropriate Penalty.	52
5.4	Tracking With 2 m Of Measurement Noise at 300 Sec.	52
5.5	Tracking With .5 m Of Measurement Noise at 300 Sec.	53
5.6	Tracking With 2 m Of Feedback Noise at 10 Sec.	53
5.7	Control Architectures	55
5.8	Implemented Decentralized Architecture	56
6.1	Hardware Setup.	58
6.2	Orion Receivers.	59
6.3	Software Flowchart.	64
7.1	Ensemble Mean Absolute Position Actual and Formal Errors. .	69
7.2	Ensemble Mean Absolute Velocity Actual and Formal Errors. .	69
7.3	1 km Ensemble Mean Relative Position	70
7.4	1 km Ensemble Mean Relative Velocity	70
7.5	100 km Ensemble Mean Relative Position	71
7.6	100 km Ensemble Mean Relative Velocity	71
7.7	500 km Ensemble Mean Relative Position	72
7.8	500 km Ensemble Mean Relative Velocity	72
7.9	Common View GPS Satellites.	74
7.10	1 km Ensemble Mean Relative Position	75
7.11	1 km Ensemble MeanRelative Velocity	76
7.12	100 km Ensemble Mean Relative Position	76

7.13	100 km Ensemble Mean Relative Velocity	77
7.14	500 km Ensemble Mean Relative Position	77
7.15	500 km Ensemble Mean Relative Velocity	78
7.16	Best Case 1 km Relative Position	81
7.17	Best Case 1 km Relative Velocity	81
7.18	Best Case 100 km Relative Position	82
7.19	Best Case 100 km Relative Velocity	82
7.20	Best Case 1 km Relative Position	83
7.21	Best Case 1 km Relative Velocity	83
7.22	Best Case 100 km Relative Position	84
7.23	Best Case 100 km Relative Velocity	84
7.24	Cycle Slip Visualization.	85
7.25	Best Case Cycle Slip Profile.	85
7.26	GPS Clock Offset Estimation.	87
7.27	GPS Frequency Offset Estimation.	88
7.28	Ensemble Mean Relative Position	90
7.29	Ensemble Mean Relative Velocity	90
7.30	Ensemble Mean Tracking Error with 300 Sec Window	91
7.31	Ensemble Mean Actuation Effort in the $\hat{u}, \hat{v}, \hat{w}$ frame	91
7.32	Full Simulation Environment Ensemble Mean Relative Position	93
7.33	Full Simulation Environment Ensemble Mean Relative Velocity	94
7.34	Ensemble Mean Relative Position	96
7.35	Ensemble Mean Relative Velocity	96
7.36	Ensemble Mean Tracking Error	97
7.37	Ensemble Mean Actuation Effort	97
C.1	Measurement Residuals	122
C.2	Absolute Position Covariances Updates	124
C.3	Relative State Covariance Updates	124

Chapter 1

Introduction

1.1 Background

Formation flying can be defined as the use of two or more vehicles moving with specified dynamics for scientific, military, or commercial purposes. Specific examples could be the docking of the Space Shuttle to the Space Station, or the Space Shuttle capturing the Hubble Space Telescope. While there have been numerous examples of formations with humans in-the-loop, the number of unmanned satellite missions have been relatively few. The disparity between these two mission types is due, in part, to the high cost of manufacturing and maintaining multiple satellites.

In recent years there has been substantial interest in autonomous satellite formations, driven by the new technologies that enable smaller and cheaper spacecraft. The advantages of using formations are numerous, and include the ability to distribute instruments among several spacecraft, thus eliminating the possibility of a single point failure. Additionally, by placing sensors on multiple platforms, new classes of missions that require simultaneous sensing from different angles become possible. Table 1.1 lists several formation missions that have been considered by the National Aeronautics and Space Administration (NASA). Two of the missions listed, TECHSAT-21 and the AQUA mission, are designed to be Earth observing formations whose mission plans lie in tandem with the work documented in this paper.

Table 1.1: Actual and Proposed Formation Flying Missions.[22]

Projected Launch	Mission
2000	Gravity Recovery and Climate Recovery (GRACE)*
2004	Techsat-21/AFRL
2004	ESSP-3-Cena (w/ Aqua)
2006	Magnetosphere Imaging Constellation (MAGIC)
2008	Laser Interferometer Space Antenna (LISA)
2011	Terrestrial Planet Finder (TPF)
15+	Planet Imager (PI)

* Launched in March 2002

The sophistication and complexity of the missions grow as one descends the time line of Table 1.1. To ensure proper functionality of these formations, our simulation ability must keep pace with the satellite design technology. This work is designed to open the door to new formation testing strategies that will accommodate flight hardware in a real-time setting.

1.1.1 TECHSAT-21

The Technology Satellite of the 21st Century (TECHSAT-21) is a formation being driven by the Air Force Research Laboratory (AFRL). The Air Force hopes to fully take advantage of the benefits of formation flying by constructing an array of small, lightweight satellites that can be reconfigured to allow for multiple mission profiles. One of the keys to the mission is having several platforms flying in a precise formation providing simultaneous measurements from multiple positions [25]. While the proof-of-concept mission may only consist of three satellites, the proposed Techsat-21 mission will be

comprised of 35 clusters of 8 satellites each [17]. The simulation system demonstrated in this work would be ideal for the modelling of a control system with real-time hardware integration.

1.1.2 A-TRAIN

The “A-Train” is an Earth observing formation that will consist of five satellites: Aqua, CALIPSO (Cloud-Aerosol Lidar and Infrared Pathfinder Satellite Observations), CloudSat, Parasol (Polarization and Anisotropy of Reflectances for Atmospheric Sciences coupled with Observations from a Lidar), and Aura. Mission constraints require an in-line formation with intersatellite distances ranging from 100 km out to 6000 km [23]. The long baseline relative navigation filter documented in this work could provide a good starting point for developing a formation-wide autonomous control system.

1.2 Previous Work

The number of formation flying papers has increased dramatically over the past five years. There are also numerous works available detailing the relative navigation problem, which has been studied for decades. This section focuses on some of the more recent works that are directly relevant to this thesis.

Several papers have been written outlining the basic fundamentals of formation flying. Bauer defines many of the key developments required for formations, including sensor development and formation control strategies [2]. Sabol expands on this work by defining some basic satellite formation types and the corresponding vehicle dynamics [24].

Ebinuma demonstrated the use of GPS for rendezvous by developing a real-time hardware-in-the-loop feedback control system at the University of Texas at Austin. The filter in this work estimated the absolute state of the each satellite, and then differenced these states to generate the relative separation. His development of the mathematical models is rigorous [9]. Ebinuma, Bishop, and Lightsey published a condensed account of these results[10].

In another work, Ebinuma, Montenbruck, and Lightsey demonstrated 20 cm relative navigation accuracies for a pair of satellites separated by 10 km [11]. They found that at this separation distance, only minimal gains were achieved when their Kalman filter employed a 10x10 gravity model as opposed to a simple model based upon Hill's equations.

Binning used single-differenced carrier phase measurements and an integer ambiguity search algorithm to explore navigation accuracies for 5 and 50 km baselines. His hardware-in-the-loop simulations generated an accuracy of 3 cm over a 50 km baseline using a dual frequency widelane integer resolution method. [3].

Wolfe and Speyer examined the relative navigation problem with vehicle separations near 100 km. They used differential carrier phase measurements coupled with a Wald test to aid in the resolution of the integer ambiguities. Their procedure yielded accuracies near 5 cm for a GPS carrier phase signal with 1 cm of noise [29].

Gramling discusses the extension of formation flying algorithms from low Earth orbits to high Earth orbits using GPS signals when available. The algorithms detailed describe relative position estimation by differencing point solutions as well as by differencing filtered solutions [14]. Further work by Long compares accuracies in relative navigation for eccentric medium and

high Earth orbits. Their filter uses GPS measurements, cross link data, and celestial object measurements to compute the states [19].

How has published several papers tackling the fundamental issues behind formation design and control. Busse and How demonstrated sub-centimeter navigation accuracy for a 1 km baseline using an adaptive Kalman filter and differential carrier phase measurements [5].

Several papers were written in conjunction with the EO-1/Landsat-7 mission. Folta provides a good overview of the mission as well as the results from the Autocon flight control system [13].

1.3 Research Contributions

The primary objective of this research was to construct a simulation tool that could be used to evaluate estimation and control systems for large formations in real-time utilizing actual hardware. In addition to the simulation environment, a basic Kalman filter was developed for use in a wide variety of satellite formations. A simple linear-quadratic control system was also employed in order to close the loop.

1.3.1 Small Formation to Large Formation Generalization

In order to understand the differences between the small and large formation cases, research was done to determine the strengths and weaknesses of both types. A literature review was conducted in order to determine the best way to construct a general Kalman filter that could leverage the simplicity of the small formation plant against the increasingly complex relative dynamics encountered with longer baselines.

1.3.2 Extended Kalman Filter Design

An extended Kalman filter (EKF) was constructed for use in the Extended Formation Flying Testbed (EFFTB) developed in this work. The filter uses the local Global Positioning System (GPS) measurements for its own absolute state estimate, and measurements from a distant receiver for relative navigation. The filter utilizes all three GPS observables: the pseudorange, the carrier phase, and the Doppler shift. To reduce the effect of common error sources, the relative measurements are all double-differenced prior to processing. Special care was taken to include error sources such as the differential ionospheric delay and the differential transmit time between receivers. These error sources have a greater impact on missions with longer baselines.

1.3.3 Demonstration of a Generalized Kalman Filter

The Kalman filter design was tested over a variety of baselines ranging from 1 km to 500 km. The 1 km simulation yielded sub-centimeter accuracies, similar to works by Ebimuma and Binning [11] [3]. By slightly shifting the gains, the filter exhibited 3-4 centimeter steady-state error levels at 100 km spacing, comparable to Speyer's work [29]. It was determined that the 500 km baseline scenario exceeded the threshold of the filter's ability due to the breakdown of the models implemented. By implementing a simple gain-scheduling algorithm into the flight computer, this single filter could allow for a group of satellites launched together to separate from the launch vehicle and position themselves up to 400 km away with less than 50 cm of error.

1.3.4 Construction of the Extended Formation Flying Testbed

A real-time, closed-loop, hardware in-the-loop simulation tool was developed to model formations with several satellites. Initially, this tool consisted of a generic communications package that could accept any number of hardware or software measurement sources. The observer and controller portions of the code were added as subroutines, making it easy to swap in and out the different modules. This allows for rapid implementation of different estimator and controller combinations. The EFFTb could be used for Low Earth Orbits (LEO), High Earth Orbits (HEO), or even deep space missions, with the appropriate propagator and measurement source substitutions.

1.3.5 GPS Specific Instance of the EFFTb

The general EFFTb architecture was adapted to use Spirent GPS simulators and Orion type GPS receivers in order to simulate a large LEO formation. The propagation machine was configured to feed the simulator reference data at 10Hz, and subroutines were developed to read in and process the GPS data correctly. The testbed was connected to a similar configuration of machines at Goddard Space Flight Center (GSFC) to generate a multiple satellite formation transferring real-time measurements over the Internet.

1.4 Overview

Chapter 2 details the different time systems and reference frames used in this work, and includes the mathematical descriptions on how to convert back and forth between these systems.

Chapter 3 describes the GPS measurement models and differentiates

between the measurements used for the absolute and relative navigation portions. The GPS satellite propagation models are also outlined.

Chapter 4 presents the extended Kalman filter equations and the dynamic models used.

Chapter 5 diagrams the equations and implementation of the linear-quadratic controller.

Chapter 6 outlines the hardware and software development of the Extended Formation Flying Testbed.

Chapter 7 describes the formation considered for this study, and presents the results of the real-time simulation environment. Preliminary results validating the filter and controller operation, as well as the validation of the testbed architecture, are also presented.

Chapter 8 summarizes the research and includes conclusions and topics for possible future work.

Chapter 2

Reference Systems

2.1 Time Systems

There are three basic time systems utilized in this work. Coordinated Universal Time (UTC) is the basic time used for computations and integration. In order to convert from the Earth Centered Inertial (ECI) coordinate frame and the Earth Centered Earth Fixed (ECEF), Greenwich Sidereal Time (GST) is used. Finally, GPS time is used in conjunction with the corresponding ephemerides to propagate the GPS satellites forward in time. The following is a short list defining each system and the transformation equations.

2.1.1 Solar and Sidereal Times

Universal time (UT) is a solar time system defined by the Greenwich Hour Angle (GHA) augmented by 12 hours of a fictitious sun uniformly orbiting in the equatorial plane [15]. Sidereal time is defined by the hour angle of the vernal equinox, and the sidereal time associated with the Greenwich Meridian is called the Greenwich Sidereal Time [27]. Since the Earth's rotation rate is not constant, neither of these time systems are uniform. In order to correct for some of this non-uniformity, the UT1 time system was introduced, which accounts for the polar motion of the Earth.

2.1.2 Atomic time

In order to meet the demands of the scientific community, the atomic time standard was introduced. International Atomic Time (IAT) is based on the quantum transitions of the Cesium-133 atom. The atomic second is based upon a fixed number of these cycles.

2.1.3 Coordinated Universal Time

In 1972, UTC was introduced to tie the irregularity of the UT1 system to the rigidity of the IAT system. In order to align the two systems, leap seconds are added to both the UTC and IAT. UTC always differs from IAT by an integer number of seconds, such that

$$IAT = UTC + 1.000n \text{ seconds} \quad (2.1)$$

The United States Naval Observatory (USNO) is tasked with the determination of this integer n , as well as the UT1-UTC corrections, and publishes them in the International Earth Rotation Service (IERS) Bulletin-A.

2.1.4 GPS Time

GPS time is an atomic time system that is based on the number of weeks elapsed from the GPS epoch, as well as the number of seconds into the current GPS week. The GPS standard epoch date is January 6, 1980 at midnight. GPS time is related to the IAT time system by a nominal 19 second offset such that:

$$IAT = GPS + 19.000 \text{ seconds} \quad (2.2)$$

Substituting equation 2.2 into 2.1 results in:

$$UTC = GPS + 19.000 - 1.000n \text{ seconds} \quad (2.3)$$

According to the USNO, in 2003 the integer value was $n = 32$, which yields a difference between UTC and GPS time of 13 seconds.

2.1.5 GPS Roll Over

In order to accurately convert between GPS time and other time systems, the GPS week must be accounted for. The week number starts at the standard GPS epoch and is incremented until it reaches 1024, at which time it is reset to 0. The first rollover occurred at midnight on August 21, 1999. The GPS week number can be obtained from the first subframe of the GPS navigation message.

2.2 Reference Frames

There are three different references systems used throughout this work. The following subsections define those system and provide the appropriate transformations.

2.2.1 Earth Centered Inertial Reference Frame (ECI)

The J2000 reference system is used predominantly throughout this dissertation. This system is a geocentric equatorial frame defined at the epoch January 1, 2000. The reference plane is constructed from the mean equatorial plane of the Earth at the J2000 epoch. All of the orbit propagation and the generation of the state transition matrix is performed in this frame. Table 2.1 describes the specifics of the coordinate system.

Table 2.1: J2000 Reference System Definition

X Axis	Mean vernal equinox at epoch
Y Axis	Normal to x and z to form a right-handed system
Z Axis	Normal to the mean equatorial plane of J2000 in the direction of the Earth's mean spin

2.2.2 Earth Centered Earth Fixed Reference Frame (ECEF)

The ECEF frame differs from the ECI frame in that it is fixed in the Earth, and the primary axis is always aligned with the Greenwich Meridian. Since it is rotating, it is not an inertial frame. The ECEF frame is used for communicating with the GPS signal generator. Table 2.2 describes the ECEF system unit vectors.

Table 2.2: ECEF Reference System Definition

X Axis	Intersection of the Greenwich Meridian with the equator
Y Axis	Normal to x and z to form a right-handed system
Z Axis	Adopted geographic pole direction

2.2.3 Spacecraft Centered Frame

The U,V,W frame is a spacecraft centered frame that is used only for presenting results. Table 2.3 describes the coordinate system. The unit vectors comprising the UVW frame, denoted \hat{u} , \hat{v} , and \hat{w} can be generated from a J2000 position using equations 2.4.

$$\begin{aligned}\underline{\hat{u}} &= \frac{\underline{r}}{|\underline{r}|} \\ \underline{\hat{v}} &= \underline{\hat{w}} \times \underline{\hat{u}}\end{aligned}\tag{2.4}$$

$$\underline{\hat{w}} = \frac{\underline{r} \times \dot{\underline{r}}}{|\underline{r} \times \dot{\underline{r}}|}$$

where:

\underline{r} = the position vector in the J2000 frame

$\dot{\underline{r}}$ = the velocity vector in the J2000 frame

Table 2.3: UVW Reference System Definition

U Axis	Points along the radius vector to the satellite
V Axis	Normal to U and W to complete a right hand system
W Axis	Direction of the orbital angular momentum

Chapter 3

GPS Models and Measurements

3.1 GPS Observables

There are three main observables provided by the GPS system, and all three are utilized within this work. This section details the pseudorange, carrier phase, and doppler measurements and the errors associated with them.

3.1.1 Pseudorange Measurement

The pseudorange is simply the range between the GPS receiver and the GPS satellite with offsets due to clock uncertainties and other measurement error sources. For the measurement equations, superscripts will be used for the GPS satellites, and subscripts will indicate the GPS receiver.

Since the true time is not known, the satellite transmit time (t^S) and the receiver acquire time (t_R) are given by:

$$t^S = T^S + \Delta t^S \quad (3.1)$$

$$t_R = T_R + \Delta t_R \quad (3.2)$$

where the capital T's are indicative of the true times, and Δt^S and Δt_R denote the satellite and receiver clock error respectively. The pseudorange measurement is simply the speed of light C times the difference between the receiver clock and the satellite clock.

$$P(t_R) = C(t_R - t^S) \quad (3.3)$$

Substituting equations 3.1 and 3.2 into 3.3 and rearranging yields:

$$P(t_R) = C(T_R - T^S) - C(\Delta t_R - \Delta t^S) \quad (3.4)$$

The first term on the right hand side of the equality is simply the true range between the GPS satellite and receiver, which will be denoted as ρ_R^S . Equation 3.4 simplifies to:

$$P(t_R) = \rho_R^S - C(\Delta t_R - \Delta t^S) \quad (3.5)$$

The measurement generated in the receiver is at the acquired time t_R not the true time T_R . Since the true time is not known, the geometric distance is linearized about the acquired time by [9]:

$$\begin{aligned} \rho_R^S &= \rho_R^S(t_R - \Delta t^R) \\ &= \rho_R^S(t_R) - \dot{\rho}_R^S(t_R)\Delta t_R \end{aligned} \quad (3.6)$$

Inserting this result into the equation 3.5 gives:

$$P(t_R) = \rho_R^S(t_R) + (C - \dot{\rho}_R^S)\Delta t_R - C\Delta t^S \quad (3.7)$$

Equation 3.7 is the mathematical equation for the pseudorange in a vacuum with no other errors. When these additional error sources are accounted for, the equation becomes:

$$P(t_R) = \rho_R^S + (C - \dot{\rho}_R^S)\Delta t_R - C\Delta t^S + \Delta_E + \Delta_{Iono} + \xi \quad (3.8)$$

where:

$$\begin{aligned} \Delta_E &= \text{User Range Error} \\ \Delta_{Iono} &= \text{Ionospheric Delay} \\ \xi &= \text{Pseudorange measurement noise} \end{aligned}$$

Three possible error sources have been omitted from the equation, but are documented here for completeness. Since this work only examines spacecraft orbits, the tropospheric delay can be neglected. Selective ability, the slewing of the GPS clock and ephemeris data, was turned off on May 1, 2000, and is therefore not accounted for. Finally the multipath error is not accounted for, as this error is highly dependent on the application in which the receiver is used.

3.1.1.1 User Range Error

The user range error is a combination of two error sources: the GPS satellite clock correction error, and GPS satellite ephemeris error. Each GPS satellite has an atomic clock on board for precise timekeeping. However, there are slight drifts and biases to these clocks which are tracked by the GPS ground segment. These offsets are broadcast in the form of curve fit coefficients in the GPS message. However, since only terms through the first order are broadcast, there is a slight error due to truncation.

The GPS ground segment also provides the GPS satellite ephemerides which are broadcast in the almanac file. Slight errors in the satellite's orbital elements results in satellite position errors when propagated.

3.1.1.2 Ionospheric Delay

The ionosphere, which extends from approximately 50 km to 1000 km above the Earth's surface, acts as a dispersive medium with respect to the GPS signals. The net affect on the measurement is that the pseudorange mea-

measurements are delayed, and therefore measured too long. The opposite affect is observed the carrier phase measurements. Due to the variable nature of the ionosphere, precise modelling of these group delays and advances is difficult.

3.1.2 Carrier Phase Measurement

The carrier phase measurement is a direct measurement of the phase of the received signal. If we denote the phase of the received carrier signal as φ^S and the phase of the reference carrier signal generated by the receiver as φ_R , we can write [15]:

$$\varphi_R(t_R) = \varphi_R(T_R) + \dot{\varphi}_R \Delta t_R \quad (3.9)$$

$$\varphi^S(t^S) = \varphi^S(T^S) + \dot{\varphi}^S \Delta t^S \quad (3.10)$$

The time t is an epoch reckoned from an initial epoch $t_0 = 0$. The carrier beat phase $\varphi_R^S(t_R)$ can be defined as:

$$\begin{aligned} \varphi_R^S(t_R) &= \varphi^S(t^S) - \varphi_R(t_R) \\ &= (\varphi^S(T^S) - \varphi_R(T_R)) + f \Delta t^S - f \Delta t_R \end{aligned} \quad (3.11)$$

where the L1 carrier frequency $\dot{\varphi}$ has been replaced with f , and the appropriate terms from equations 3.9 and 3.10 have substituted in. The term f/C can be used to convert the geometric distance ρ into cycles. Doing so allows the true

beat phase to be represented by:

$$\begin{aligned}\varphi^S(T^S) - \varphi_R(T_R) &= -\frac{f}{C}\rho_R^S(T_R) \\ &= -\frac{f}{C}(\rho_R^S(t_R) - \dot{\rho}_R^S(t_R)\Delta t_R)\end{aligned}\quad (3.12)$$

Substituting 3.12 into the first term in 3.11 gives the beat phase

$$\varphi_R^S(t_R) = -\frac{f}{C}\rho_R^S(t_R) - f\left[1 - \frac{\dot{\rho}_R^S(t_R)}{C}\right]\Delta t_R + f\Delta t^S \quad (3.13)$$

When the receiver is switched on the initial epoch is set, and the beat phase is measured. The receiver is unable to determine the integer number of whole phase cycles between the receiver and GPS satellite, only the portion of the cycle it has measured. This cycle ambiguity N , remains a constant as long as signal lock is not lost to the GPS satellite. Therefore, at some epoch t , the beat phase can be represented by:

$$\varphi_R^S(t) = \Delta\varphi_R^S|_{t_0}^t + N \quad (3.14)$$

The term $\Delta\varphi_R^S$ represents the phase measured between the times t and t_0 . The carrier phase measurement can be realized by substituting 3.13 into 3.14, denoting the observation as Φ , and assigning it a negative sign:

$$\Phi_R^S(t_R) = \frac{f}{C}\rho_R^S(t_R) + f\left(1 - \frac{\dot{\rho}_R^S(t_R)}{C}\right)\Delta t_R - f\Delta t^S + N \quad (3.15)$$

The carrier phase measurement is often represented in terms of range, so we can scale it by $\lambda = \frac{C}{f}$ where λ is the carrier wavelength.

$$\lambda\Phi_R^S(t_R) = \rho_R^S(t_R) + (C - \dot{\rho}_R^S(t_R))\Delta t_R - C\Delta t^S + \lambda N \quad (3.16)$$

As with the pseudorange equation, when the error sources are added in, the carrier phase becomes:

$$\lambda\Phi_R^S(t_R) = \rho_R^S(t_R) + (C - \dot{\rho}_R^S(t_R))\Delta t_R - C\Delta t^S + \lambda N + \Delta_E - \Delta_{Iono} + \zeta \quad (3.17)$$

where ζ is the carrier phase measurement noise. It is worth noting that the only qualitative differences between the pseudorange measurement and the carrier phase measurement is the addition of the integer ambiguity term at the end, and the opposite sign of the ionosphere delay.

3.1.3 Doppler Measurement

Since the transmitting satellite in the GPS constellation is always in motion relative to the receiving body, the received frequency will be Doppler shifted. In other words, the received frequency will differ from the transmitted frequency due to the relative motion. The equation for the Doppler measurement scaled to units of range can be obtained by differentiating equation 3.17 to obtain:

$$D(t) = \lambda\dot{\Phi} = \dot{\rho} + C(\Delta f_R - \Delta f^S) - \dot{\Delta}_{Iono} + \delta \quad (3.18)$$

Where δ term is the noise in the Doppler measurement. The rate of change of the ionospheric delay is often neglected in practical work.

3.2 Double-Differenced Observables

Many of the error sources described above are common mode, and will cancel out when differenced, leaving a more accurate relative measurement. For this reason, double-differenced carrier phase and doppler measurements are utilized for the relative positioning portion of the filter. A schematic of the double-difference process is diagramed in Figure 3.1. In the figure, SAT

A receives measurements from GPS A and GPS B at the same epoch. SAT A subtracts these measurements, creating a single-difference. At the same epoch, SAT B performs the same single-difference. The double-difference is performed by subtracting the two single-differences. The double-differenced carrier phase and doppler measurements will be developed mathematically in the following subsections.

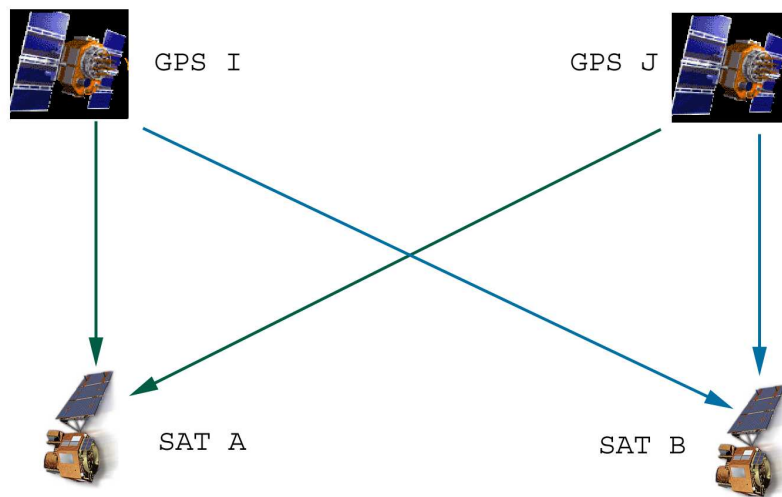


Figure 3.1: Double-Difference.

3.2.1 Double-Differenced Carrier Phase

If the carrier phase measurement between SAT A and GPS I is denoted as Φ_A^I , the measurement can be represented by:

$$\lambda\Phi_A^I(t) = \rho_A^I + (c - \dot{\rho}_A^I)\Delta t_{R,A} - C\Delta t^{S,I} + \Delta_{E,I} + \Delta_{Iono,IA} + \lambda N_A^I + \zeta_I$$

A similar equation can be written for SAT A and GPS J :

$$\lambda\Phi_A^J(t) = \rho_A^J + (c - \dot{\rho}_A^J)\Delta t_{R,A} - C\Delta t^{S,J} + \Delta_{E,J} + \Delta_{Iono,JA} + \lambda N_A^J + \zeta_J$$

Subtracting these equations yields the single difference equation for SAT A:

$$\lambda\Phi_A^{IJ} = (\rho_A^I - \rho_A^J) - C\Delta t^{S,I} + C\Delta t^{S,J} + (\Delta_{Iono,IA} - \Delta_{Iono,JA}) + (\lambda N_A^I - \lambda N_A^J) + (\zeta_A^I - \zeta_A^J)$$

The receiver clock error terms have dropped out due to the differencing. A similar equation can be written for the SAT B receiver:

$$\lambda\Phi_B^{IJ} = (\rho_B^I - \rho_B^J) - C\Delta t^{S,I} + C\Delta t^{S,J} + (\Delta_{Iono,IB} - \Delta_{Iono,JB}) + (\lambda N_B^I - \lambda N_B^J) + (\zeta_B^I - \zeta_B^J)$$

performing the second differencing yields:

$$\begin{aligned} \lambda\Phi_{AB}^{IJ} = & ((\rho_A^I - \rho_A^J) - (\rho_B^I - \rho_B^J)) + ((\Delta_{Iono,IA} - \Delta_{Iono,JA}) - (\Delta_{Iono,IB} - \Delta_{Iono,JB})) \\ & + (\lambda N_A^I - \lambda N_A^J) - ((\lambda N_B^I - \lambda N_B^J)) + \gamma \end{aligned}$$

where γ is the double differenced random errors. The GPS satellite clock errors and the user range errors have dropped out, leaving the double differenced ionosphere as the sole non-random error. For short intersatellite distances, the path between the GPS satellites and the receivers are close enough that this error is negligible.

3.2.2 Double-Differenced Doppler

Following the above derivation, the Doppler equation is written for the measurement between SAT A and GPS I.

$$D_A^I(t) = \dot{\rho}_A^I + C(\Delta f_{R,A} - \Delta f^{S,I}) - \dot{\Delta}_{Iono,A}^I + \delta_A^I \quad (3.19)$$

For the SAT A and GPS J combination:

$$D_A^J(t) = \dot{\rho}_A^J + C(\Delta f_{R,A} - \Delta f^{S,J}) - \dot{\Delta}_{Iono,A}^J + \delta_A^J \quad (3.20)$$

Taking the single-difference yields:

$$D_A^{IJ}(t) = (\dot{\rho}_A^I - \dot{\rho}_A^J) - C(\Delta f^{S,I} + \Delta f^{S,J}) - (\dot{\Delta}_{Iono,A}^I - \dot{\Delta}_{Iono,A}^J) + (\delta_A^I - \delta_A^J)$$

Writing the complementary equation for SAT B and subtracting gives the double-differenced Doppler equation:

$$\begin{aligned} D_{AB}^{IJ}(t) = & ((\dot{\rho}_A^I - \dot{\rho}_A^J) - (\dot{\rho}_B^I - \dot{\rho}_B^J)) - ((\dot{\Delta}_{Iono,A}^I - \dot{\Delta}_{Iono,A}^J) \\ & - (\dot{\Delta}_{Iono,B}^I - \dot{\Delta}_{Iono,B}^J)) + (\delta_A^I - \delta_A^J) - (\delta_B^I - \delta_B^J) \end{aligned}$$

3.2.3 GPS Satellite Propagation

In order to predict the location of the GPS satellites, the GPS receiver uses the almanac data which is superimposed upon the GPS signal. The almanac data is updated at least every six days and contains orbital ephemerides and satellite clock corrections. Table 3.1 provides an overview of the almanac data elements [15].

The propagation of the elements from the almanac is straightforward, and the ECEF satellite coordinates at time t can be readily obtained using the following algorithm [20].

Table 3.1: Almanac Information

Parameter	Description
ID	Satellite PRN number
Week	Current GPS week
t_a	Reference epoch in seconds within current GPS week
\sqrt{a}	Square root of the semi major axis
e	Eccentricity
M_0	Mean anomaly at reference epoch
ω	Argument of Perigee
i	Inclination
ℓ	Longitude of the node at the weekly epoch
$\dot{\Omega}$	Drift of the right ascension of the node
a_0	Satellite clock offset
a_1	Satellite clock drift

Determine the instantaneous longitude of the ascending node:

$$\Omega = \Omega_0 + \dot{\Omega}(t - t_a) - \omega_E(t - t_0)$$

where t_0 denotes the start of the current GPS week, and ω_E is the rotation rate of the Earth. Solve for the eccentric anomaly E using Kepler's equation

$$E - e \sin E = M_0 + \sqrt{\frac{GM_E}{a^3}}(t - t_a) \quad (3.21)$$

The ECEF position is obtained via the following rotations:

$$\underline{r}_{ECEF} = R_z(-\Omega)R_x(-i)R_z(-\omega) \begin{bmatrix} a(\cos(E) - e) \\ a\sqrt{1 - e^2}\sin(E) \\ 0 \end{bmatrix} \quad (3.22)$$

where:

$$R_z(-\Omega) = \begin{bmatrix} \cos(-\Omega) & \sin(-\Omega) & 0 \\ -\sin(-\Omega) & \cos(-\Omega) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_x(-i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-i) & \sin(-i) \\ 0 & -\sin(-i) & \cos(-i) \end{bmatrix}$$

$$R_z(-\omega) = \begin{bmatrix} \cos(-\omega) & \sin(-\omega) & 0 \\ -\sin(-\omega) & \cos(-\omega) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Special care must be taken to ensure that the geometric range used in the measurement equations is the range generated using the receiver position at t_R and the GPS satellite position at time t^S . This is accomplished by applying the following iterative scheme [9].

The signal transmission time is solved using:

$$t^{S,n+1} = t_R - \frac{|\underline{r}^S(t^{S,n}) - \underline{r}_R(t_R)|}{C}$$

where \underline{r}^S , \underline{r}_R are the GPS and receiver states respectively and $t^{S,n}$ is the n th approximation for t^S , the signal transmit time from the GPS satellite. The iteration continues until:

$$|t^{S,n+1} - t^{S,n}| < \epsilon$$

The default value for the tolerance ϵ is 1e-12 which yields less than 1 mm of accuracy for the range measurement. Usually no more than 3 iterations are required for convergence.

Chapter 4

Kalman Filter Design

4.1 General Formation Description

The design of any particular formation is mission dependent, and several very interesting formation designs have been developed. In order to generalize this work, it is assumed that each satellite will generate a relative navigation solution with respect to one common satellite. This vehicle, which will be labelled the master satellite, shares its measurements with each of the remaining satellites, which are designated as nodes[5]. The nodes use these measurements to generate a relative state estimate at each epoch. While this assumption may be impractical in some missions, many formations, such as a multi-vehicle space telescope, emphasize the relative distances from some central craft.

One of the advantages in this scenario is that, for estimation purposes, the formation flying problem breaks down into several relative navigation problems that can be examined independently. As described in Section 4.6, all of the measurement equations are rewritten to remove the necessity of estimating the absolute position of the hub, though its state is reconstructed from the node state and the relative position at every epoch. Figure 4.1 details the layout of the absolute and relative vectors.

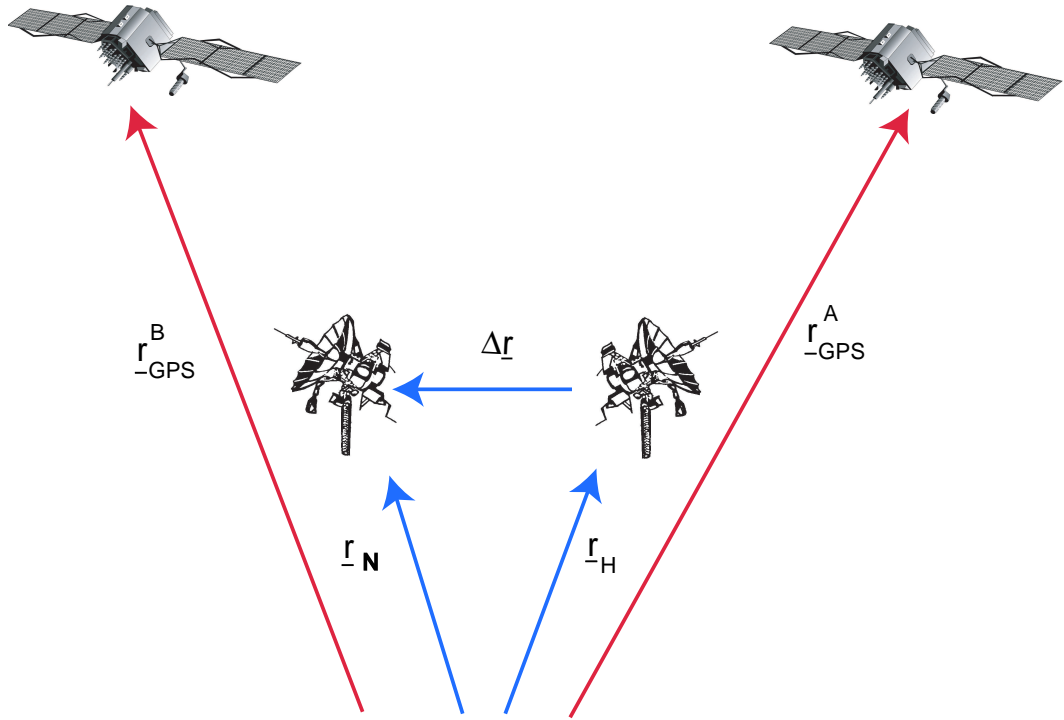


Figure 4.1: Problem Setup.

4.2 Filter State

The estimation state vector \underline{X} at each node can be written as:

$$\underline{X} = \begin{bmatrix} r \\ \Delta r \\ b \\ \dot{b} \\ \lambda N \\ B \end{bmatrix}$$

where:

- \underline{r} = Absolute state of the node
- $\Delta \underline{r}$ = Relative state between the node and hub
- b = Receiver clock offset
- \dot{b} = Receiver clock frequency offset
- λN = Double-Differenced integer ambiguities
- B = GPS clock offset

Each of the states are developed more completely in the following subsections.

4.2.1 Absolute And Relative States

Each node satellite estimates its own absolute state as well as the relative state to the hub:

$$\begin{bmatrix} \underline{r} \\ \Delta \underline{r} \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \Delta x \\ \Delta y \\ \Delta z \\ \Delta \dot{x} \\ \Delta \dot{y} \\ \Delta \dot{z} \end{bmatrix}$$

where each of these elements are realized in the ECI reference frame. For simplicity, the GPS receiver is assumed to be located at the center of mass of each satellite. While estimation of the attitude is crucial to a formation mission, it will be left for future research.

4.2.2 Receiver Clock and Frequency Offset State

The GPS receiver typically has a quartz crystal clock on board, making it much less accurate than the ones in the GPS satellites. In order to compensate for these clock errors in the pseudorange and doppler measurements, the receiver clock and frequency offsets must be estimated. This may be achieved by means of a second order Gauss-Markov process.

The Orion GPS receivers used for this research are enabled with a clock-steering algorithm. This code slightly changes the TIC, the receiver's fundamental time unit, so that the GPS receiver is always aligned to the GPS time. This makes exchanging measurements much more feasible. Since the receiver clock is no longer drifting freely, the commonly used receiver clock model no longer applies. Instantaneous changes in the receiver clock are not reflected in the clock's frequency, creating a situation where the two states vary independently. It was empirically determined that the clock error could be successfully modelled using a constant offset b and a constant frequency offset \dot{b} . These states are modelled as uncorrelated random walk processes.

4.2.3 Double-Differenced Ambiguity State

The double-differenced ambiguity state is defined by:

$$\lambda N = \begin{bmatrix} \lambda N_{AB}^{12} \\ \lambda N_{AB}^{13} \\ \vdots \\ \lambda N_{AB}^{1(n-1)} \end{bmatrix}$$

where n is the number of GPS satellites tracked by both GPS receivers.

4.2.4 GPS Clock Offset

As described in Section 3.1.1.1, there is a slight error in the transmitted GPS clock model due to the truncation of the model down to the first order. This error can be accurately modelled as a constant bias for each GPS satellite tracked.

$$B = \begin{bmatrix} B_A^1 \\ B_A^2 \\ \vdots \\ B_A^n \end{bmatrix}$$

This error is common mode, and cancels out for the double-differenced measurements. Consequently, the B state is needed for the pseudorange measurement only.

4.3 Kalman Filter

The feedback control system that makes up the software portion of the EFFTb requires the implementation of an estimator and a controller. The backbone of the software is general enough to allow for many combinations of those elements. Many different estimator architectures have been developed for the relative navigation problem, such as the centralized, decentralized, and partially centralized formation. Instead of developing a filter that provides the “best” results for a single baseline length, a different approach was taken in this dissertation. A Kalman filter has been developed which yields good results over a wide variety of baselines. Thus some accuracy has been traded for a more versatile filter design.

Each satellite in the formation will run its own EKF for absolute and relative state estimation. The filter functions as a weighting scheme between the current measurement set and the dynamic model. The filtering can be

broken up into several distinct stages for clarity, and the mathematical descriptions will be developed accordingly.

4.3.1 State Vector Construction

Due to the chosen estimation strategy, the number of estimated states changes with the addition or removal of each GPS satellite. Therefore, if the number of GPS satellites tracked changes, the state vector must be resized to account for the appropriate number of tracked satellites. The error-covariance matrix must be rescaled accordingly as well. The hub state is also reconstructed from the current state parameters.

4.3.2 Propagation

The propagation step is used to bridge the gap in time between successive measurements. In an extended Kalman filter, the state is propagated by the direct integration of the non-linear dynamic equations. The state error covariance is propagated forward using a state transition matrix. If we denote $\underline{X}_k(+)$ as the state vector at time k after the measurement, and $\underline{X}_{k+1}(-)$ as the state vector at time $k+1$ before the measurement, then the propagation algorithm is presented as:

1. Propagate the position and velocity vectors for the absolute states by integrating the non-linear dynamical model, which is provided in section 4.4.1. The states are integrated using a Runge-Kutta 4th order numerical integrator.

$$\ddot{\underline{r}} = \underline{a}_E + \underline{a}_p \quad (4.1)$$

where \underline{a}_E is the acceleration due to the Earth's gravity and \underline{a}_p is the

desired perturbative forces.

2. The relative state is not propagated directly. Instead, the reconstructed hub state is propagated forward to the current epoch via Equation 4.1. The relative state is then formed by subtracting the current node state from the hub state.
3. The remaining terms in the state are modelled as constants, so their values are changed only by the inclusion of process noise during the propagation step.
4. The state error covariance matrix contains the estimates for the closeness of the fit with the actual observations [27]. The diagonal terms are the variances of the estimate of the state parameters. The square roots of the variances are the standard deviations for each element of the state space. For a properly working filter, the errors in the problem must be accurately reflected by the entries in the covariance matrix. The error covariance must be propagated during this step. The state error covariance matrix, \mathbf{P}_k at time t_k is defined by the expectation operator as:

$$\mathbf{P}_k = E \left\{ \left[\hat{\underline{X}}_k - \underline{X}_k \right] \left[\hat{\underline{X}}_k - \underline{X}_k \right]^T \right\} \quad (4.2)$$

The update equation for the covariance is given by:

$$\mathbf{P}_{k+1}(-) = \boldsymbol{\Phi}(t_{k+1,k}) \mathbf{P}_k(+) \boldsymbol{\Phi}^T(t_{k+1,k}) + \mathbf{Q}_{k+1} \quad (4.3)$$

The generation of the state transition matrix $\boldsymbol{\Phi}(t_{k+1,k})$ and the covariance propagation strategy are discussed in Section 4.4.2. The process noise matrix \mathbf{Q} is used as a tuning parameter.

4.3.3 Initialization

Each state of the EKF must be initialized before it is processed the first time. The node state is initialized from the GPS position fix solution at the initial epoch. The initial hub position is also set to its position fix solution, and the initial relative state is set to the difference between the two. The double-differenced ambiguity term is initialized as the difference between the double-differenced pseudorange and the double-differenced carrier phase measurements. The remainder of the states are initialized to zero.

4.3.4 Measurement Update

Once the state vector and error covariance matrix are propagated to the current time, the updated state vector $\underline{X}_k(+)$ and the updated covariance matrix $\mathbf{P}_k(+)$ can be computed. The update requires the construction of the measurement noise covariance matrix \mathbf{R}_k . The measurement equation matrix \mathbf{G} is covered in section 5.5.

1. The predicted measurement is generated by evaluating the non-linear measurement models along the current state:

$$(\hat{\underline{Y}}_k) = \mathbf{G}(\hat{\underline{X}}, t_k) \quad (4.4)$$

The residual vector is computed as the difference between the measurement and the predicted measurement:

$$\underline{z}_k = \underline{Y}_k - (\hat{\underline{Y}}_k) \quad (4.5)$$

where the hat denotes an estimated quantity. The \mathbf{H}_k matrix is generated by taking the partial derivative of the measurement equations with

respect to the state. These partial derivatives are presented in Appendix B.

$$\mathbf{H}_k = \frac{\partial \mathbf{G}}{\partial \underline{X}} \big|_{\underline{X}=\hat{\underline{X}}_k(-)} \quad (4.6)$$

2. The error covariance is updated using the numerically stable *Joseph formula*:

$$\mathbf{P}_k(+) = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k(-) (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T \quad (4.7)$$

where the Kalman gain is given by:

$$\mathbf{K}_k = \mathbf{P}_k(-) \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k(-) \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (4.8)$$

The state update also uses the Kalman gain:

$$\hat{\underline{X}}_k(+) = \hat{\underline{X}}_k(-) + \mathbf{K}_k \underline{z} \quad (4.9)$$

4.4 Dynamic Model and State Transition Matrix

In an EKF, the state is propagated forward using the non-linear differential equations. The covariance matrix, however, uses a state transition matrix for propagation. The following sections provides the full non-linear differential equations and the state transition matrix structure.

4.4.1 Dynamic Model

One of the potential benefits for formation flying is to be able to create several smaller, cheaper, less-capable nodes to replace large monolithic satellites. With this philosophy in mind, the dynamic model was intentionally left relatively simple, which reduces the computational burden. The acceleration

equations contain the gravitational attraction of the Earth, the J_2 zonal harmonic, and the atmospheric drag effect with an exponential air density model. These equations are detailed in Equation 4.10.

$$\begin{aligned}
\ddot{x} &= -\frac{\mu x}{r^3} \left[1 - \frac{3}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(5 \frac{z^2}{r^2} - 1 \right) \right] - \frac{1}{2} C_D \frac{A}{M} \rho V (\dot{x} + \omega_E y) \\
\ddot{y} &= -\frac{\mu y}{r^3} \left[1 - \frac{3}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(5 \frac{z^2}{r^2} - 1 \right) \right] - \frac{1}{2} C_D \frac{A}{M} \rho V (\dot{y} - \omega_E x) \\
\ddot{z} &= -\frac{\mu z}{r^3} \left[1 - \frac{3}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(5 \frac{z^2}{r^2} - 3 \right) \right] - \frac{1}{2} C_D \frac{A}{M} \rho V \dot{z}
\end{aligned} \tag{4.10}$$

where ω_E is the rotation rate of the Earth, R_E is the Earth's radius, C_d is the drag coefficient of the satellite, m is the satellite's mass, and A is the instantaneous cross-sectional area. The magnitude of the relative wind, V , can be computed from:

$$V = \sqrt{\dot{x}^2 + 2\omega_E \dot{x}y + \omega_E^2 y^2 + \dot{y}^2 - 2\omega_E \dot{y}x + \omega_E^2 x^2 + \dot{z}^2} \tag{4.11}$$

The value of ρ , the atmospheric density, is generated using an exponentially varying model [27]:

$$\rho = \rho_0 e^{-(r-r_0/H)} \tag{4.12}$$

Here:

$$\begin{aligned}
\rho_0 &= 3.725 \times 10^{-12} (kg/m^3) \\
r &= \text{current altitude}(m) \\
r_0 &= \text{reference altitude} = 400000(m) \\
H &= \text{scale height} = 58515(m)
\end{aligned}$$

4.4.2 State Transition Matrix

While much of the state is propagated directly via the dynamic equations, the covariance matrix must be propagated forward through the use of a state transition matrix. The state transition matrix used to propagate between the two times t_k and t_{k+1} satisfies:

$$\begin{aligned}\dot{\Phi}(t_{k+1}, t_k) &= \mathbf{A}(t_k)\Phi(t_{k+1}, t_k) \\ \Phi(t_k, t_k) &= \mathbf{I}\end{aligned}\tag{4.13}$$

$$\mathbf{A}(t_k) = \left. \frac{\partial F}{\partial \underline{X}} \right|_{\underline{X}=\hat{\underline{X}}_k(-)}\tag{4.14}$$

$$\tag{4.15}$$

where \mathbf{I} is the appropriately sized identity matrix and F is the set of dynamical equations. The state transition matrix has the block diagonal form:

$$\Phi = \begin{bmatrix} \Phi_{\text{Abs}} & 0 & \cdots & 0 \\ 0 & \Phi_{\text{Rel}} & & 0 \\ \vdots & & \Phi_{\text{b}} & \vdots \\ 0 & & & \Phi_{\Delta N} & 0 \\ 0 & 0 & \cdots & & \Phi_B \end{bmatrix}\tag{4.16}$$

each submatrix is described in the following subsections.

4.4.2.1 Absolute State Acceleration Partial Derivatives Φ_{Abs}

The acceleration partial derivatives are found by taking the partial derivatives of equation 4.10 with respect to the state. The equations for development of the absolute partial derivatives are provided in Appendix A.

4.4.2.2 Relative State Acceleration Partial Derivatives Φ_{Rel}

The relative state is not propagated directly, and the following approximation is made in order to generate the state transition matrix. The relative state at time t_{k+1} can be written as:

$$\Delta_r(t_{k+1}) = \Phi_N(t_{k+1}, k) \underline{x}_N - \Phi_H(t_{k+1}, k) \underline{x}_H \quad (4.17)$$

If we assume that, for separations under a certain threshold, the accelerations felt by the hub spacecraft are approximately equal to those of the node spacecraft, then we can say that the relative state transition matrix block can be generated from the partial differential equations of either the node or the hub. This approximation is validated by evaluating the partial differential equations numerically and comparing them to the non-linear models. The results of this analysis are presented in Appendix C.

4.4.2.3 Receiver Clock States Partial Derivatives Φ_b

The receiver clock and frequency offsets are modelled as random-walk processes. The contribution to the state transition matrix is given by:

$$\Phi_b(t_{k+1}, t_k) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (4.18)$$

4.4.2.4 Double-Differenced Ambiguity Partial Derivatives $\Phi_{\Delta N}$

The double-differenced ambiguity terms are modelled as random-walk processes. Though theory describes these ambiguities as integers, their values are not constrained as such. Their partial derivative submatrix has the form:

$$\Phi_{\Delta N}(t_{k+1}, t_k) = I \quad (4.19)$$

For N GPS satellites, there are $(N-1)$ double-differenced ambiguity terms to estimate at every epoch.

4.4.2.5 GPS Clock Error Model Φ_B

The GPS satellite clock error is modelled as a random-walk process.

$$\Phi_{\Delta B}(t_{k+1}, t_k) = I \quad (4.20)$$

Each GPS satellite tracked requires one offset parameter.

4.5 Process Noise Matrix

As the time progresses into the simulation, the covariance matrix will begin to converge. The filter will be increasingly less sensitive to the new measurements. To overcome this effect, process noise is added in the form of the process noise matrix \mathbf{Q} . This matrix also accounts for the errors in the dynamical model. The \mathbf{Q} matrix can be represented in the block diagonal form:

$$\mathbf{Q}_k = \begin{bmatrix} Q_R & 0 & 0 & 0 \\ 0 & Q_b & & 0 \\ \vdots & & Q_B & \vdots \\ 0 & \dots & \dots & Q_N \end{bmatrix} \quad (4.21)$$

Each submatrix is described below.

4.5.1 Absolute and Relative State Process Noise

The process noise covariance matrix for the absolute and relative state is defined by:

$$\mathbf{Q}_R = \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1}, \tau) \mathbf{S}(\tau) \mathbf{\Phi}^T(t_{k+1}, \tau) d\tau \quad (4.22)$$

where $S(t)$ is the process noise spectral density matrix. If the state transition matrix can be assumed constant over the time span Δt , the process noise matrix can be written in discrete form:

$$\mathbf{Q}_R = \mathbf{S}(\tau) \quad (4.23)$$

and the absolute and relative state process noise contributions are defined as:

$$\mathbf{Q}_R = \begin{bmatrix} \mathbf{Q}_{Ra} & 0 \\ 0 & \mathbf{Q}_{Rr} \end{bmatrix} \quad (4.24)$$

where \mathbf{Q}_{Ra} and \mathbf{Q}_{Rr} correspond to the absolute state and relative state process noise respectively. These matrices can be written as:

$$\mathbf{Q}_{Ra} = \begin{bmatrix} \sigma_{x_{ABS}}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{y_{ABS}}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{z_{ABS}}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\dot{x}_{ABS}}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\dot{y}_{ABS}}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{\dot{z}_{ABS}}^2 \end{bmatrix} \quad (4.25)$$

and:

$$\mathbf{Q}_{Rr} = \begin{bmatrix} \sigma_{x_{REL}}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{y_{REL}}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{z_{REL}}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\dot{x}_{REL}}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\dot{y}_{REL}}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{\dot{z}_{REL}}^2 \end{bmatrix} \quad (4.26)$$

4.5.2 GPS Receiver Clock Process Noise

The process noise covariance for the receiver clock model is:

$$\mathbf{Q}_b = \begin{bmatrix} \sigma_b^2 & 0 \\ 0 & \sigma_{\dot{b}}^2 \end{bmatrix} \quad (4.27)$$

The σ_b^2 and $\sigma_{\dot{b}}^2$ represent the variance for the clock and frequency offsets. This model is valid only for the receiver used in this experiment, due to the decoupling of the clock states by the clock steering algorithm.

4.5.3 Double-Differenced Ambiguities Process Noise

The process noise covariance block for the double-differenced ambiguity terms have the form:

$$\mathbf{Q}_N = \mathbf{I}\sigma_N^2 \quad (4.28)$$

where \mathbf{I} is an $(n-1) \times (n-1)$ identity matrix, and σ_N^2 is the variance for the double differenced ambiguities.

4.5.4 GPS Satellite Clock Bias

The process noise covariance entries for the GPS satellite clock bias also have a diagonal form:

$$\mathbf{Q}_B = \mathbf{I}\sigma_B^2 \quad (4.29)$$

where \mathbf{I} is an $(n) \times (n)$ identity matrix, and σ_B^2 is the variance of the GPS clock bias.

4.6 GPS Measurement Equations

Sections 3.1 and 3.2 describe the basic formulation for the measurements utilized by the filter. The partial derivative of these equations, which

are required for the measurement update portion of the filter, are provided in Appendix B.

4.6.1 Undifferenced Measurement Equations

The undifferenced measurement equations are very similar to the theoretical ones presented in Section 3.1. The pseudorange filter model is:

$$P(t_R) = \rho_R^S + b + B + \Delta_{Iono} \quad (4.30)$$

where B is the scaled GPS clock state offset, b is the scaled receiver clock state offset, and Δ_{Iono} is the pseudorange ionospheric delay. The ionosphere calculation uses a simplified model which consists of a shell of constant thickness and uniform electron distribution. The delay is calculated as [26]:

$$\Delta_{Iono} = \frac{82.1TEC}{F_C^2 \sqrt{\sin^2(EL) + 0.076} + \sin^2(EL)} \quad (4.31)$$

where:

TEC = $2e17$ Constant Total Electron Count

F_C = $1575.42e6$ Frequency of the carrier Signal

EL = Elevation of the GPS satellite from a local horizon

The equation for the Doppler model is given by:

$$D(t) = \dot{\rho}_R^S + \dot{b} \quad (4.32)$$

where \dot{b} is the receiver clock frequency state.

4.6.2 Double-Differenced Measurement Equations

In order to remove the necessity of estimating the absolute state of the hub satellite, the measurement for the relative states are rewritten in terms of

the node state and the relative state. If we define our system such that:

$$\Delta \underline{r}(t) = \underline{r}_H(t) - \underline{r}_N(t) \quad (4.33)$$

then the absolute state of the hub can be represented by:

$$\underline{r}_H(t) = \Delta \underline{r}(t) + \underline{r}_N(t) \quad (4.34)$$

Plugging this expression for the hub into the double-differenced measurement equations yields the following:

$$\begin{aligned} \delta \lambda \Phi_{AB}^{IJ}(t) = & \left[(|\underline{r}_{GPS}^A(t - \tau) - \Delta \underline{r}(t) - \underline{r}_N(t)|) - (|\underline{r}_{GPS}^A(t - \tau) - \underline{r}_N(t)|) \right] - \\ & \left[(|\underline{r}_{GPS}^B(t - \tau) - \Delta \underline{r}(t) - \underline{r}_N(t)|) - (|\underline{r}_{GPS}^B(t - \tau) - \underline{r}_N(t)|) \right] - \\ & \delta \Delta_{Iono} + \delta \lambda N \end{aligned}$$

where the δ notation signifies a double-difference such that:

$$\delta \lambda N = (\lambda N_H^A - \lambda N_N^A) - (\lambda N_H^B - \lambda N_N^B) \quad (4.35)$$

Similarly, the double-differenced Doppler equation has the form:

$$\begin{aligned} \delta D_{AB}^{IJ}(t) = & (\dot{\underline{r}}_{GPS}^A(t - \tau) - \Delta \dot{\underline{r}}(t) - \dot{\underline{r}}_N(t)) \cdot \frac{(\underline{r}_{GPS}^A(t - \tau) - \Delta \underline{r}(t) - \underline{r}_N(t))}{(|\underline{r}_{GPS}^A(t - \tau) - \Delta \underline{r}(t) - \underline{r}_N(t)|)} - \\ & (\dot{\underline{r}}_{GPS}^B(t - \tau) - \Delta \dot{\underline{r}}(t) - \dot{\underline{r}}_N(t)) \cdot \frac{(\underline{r}_{GPS}^B(t - \tau) - \Delta \underline{r}(t) - \underline{r}_N(t))}{(|\underline{r}_{GPS}^B(t - \tau) - \Delta \underline{r}(t) - \underline{r}_N(t)|)} \end{aligned}$$

4.7 Measurement Noise Covariance

The measurement noise covariance matrix represents the uncertainty in the measurement model. The measurement noise covariance matrix, \mathbf{R} has the block diagonal structure:

$$\mathbf{R}_k = \begin{bmatrix} \mathbf{R}_P & 0 & 0 & 0 \\ 0 & \mathbf{R}_D & 0 & 0 \\ 0 & 0 & \mathbf{R}_{\delta \lambda \Phi} & 0 \\ 0 & 0 & 0 & \mathbf{R}_{\delta D} \end{bmatrix}$$

where $\mathbf{R}_P, \mathbf{R}_D, \mathbf{R}_{\delta\lambda\Phi}, \mathbf{R}_{\delta D}$, correspond to the pseudorange, doppler, double-differenced carrier phase, and double-differenced doppler measurements respectively.

4.7.1 Pseudorange Measurement

The raw pseudorange measurements are inherently uncorrelated, yielding a diagonal block matrix:

$$\mathbf{R}_P = \begin{bmatrix} \sigma_\xi^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_\xi^2 \end{bmatrix}$$

where σ_ξ^2 is the standard deviation of the pseudorange error.

4.7.2 Doppler Measurement

The doppler measurements are also uncorrelated, and have a similar diagonal structure:

$$\mathbf{R}_D = \begin{bmatrix} \sigma_\delta^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_\delta^2 \end{bmatrix}$$

where σ_δ^2 is the standard deviation of the doppler error.

4.7.3 Double-Differenced Carrier Phase Measurement

While the raw carrier phase measurements are uncorrelated, the double-differenced measurements are correlated with each other. If we write two

double-differenced carrier phase measurements in matrix representation:

$$\begin{bmatrix} \Phi_{NH}^{AB} \\ \Phi_{NH}^{AC} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \varphi_N^A \\ \varphi_H^A \\ \varphi_N^B \\ \varphi_H^B \\ \varphi_N^C \\ \varphi_H^C \end{bmatrix}$$

The covariance matrix would be generated by:

$$\mathbf{R}_{\delta\lambda\Phi} = \begin{bmatrix} 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \sigma_{\delta\lambda\Phi}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\delta\lambda\Phi}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\delta\lambda\Phi}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\delta\lambda\Phi}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\delta\lambda\Phi}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{\delta\lambda\Phi}^2 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 \\ 1 & -1 \\ 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix}$$

where $\sigma_{\delta\lambda\Phi}^2$ is the spectral density of the double-differenced carrier phase measurement. Performing the matrix multiplication yields:

$$\mathbf{R}_{\delta\lambda\Phi} = \sigma_{\delta\lambda\Phi}^2 \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}$$

This pattern continues as more measurements are added. The double-differenced doppler measurements are likewise correlated, and their measurement error covariance block structures must be generated with 4 on the diagonals and 2s in all other elements.

Chapter 5

Control System

The EFFTb uses a linear quadratic regulator (LQR) to maintain the desired separation between the formation satellites. The following section outlines the development and implementation of the LQR used in this work.

5.1 Controller Development

While most natural processes can be represented with continuous functions, controlling these systems is usually a discrete time process. Since an optimal controller is desired, a cost function $J(t)$ is constructed such that:

$$J(t) = \int_t^{t_f} [\underline{x}_*^T(t) \mathbf{W}(t) \underline{x}_*(t) + \underline{u}^T(t) \mathbf{V}(t) \underline{u}(t)] dt + \underline{x}_*^T(t_f) \mathbf{S}_f(t_f) \underline{x}_*(t_f) \quad (5.1)$$

The control vector $\underline{u}(t)$ is said to be optimal if it minimizes this cost function. In this equation, \underline{x}_* is the difference between the current trajectory and some reference trajectory. The matrices $\mathbf{W}(t)$, $\mathbf{V}(t)$, and $\mathbf{S}_f(t)$ represent the state penalty matrix, the control penalty matrix, and the terminal penalty matrix respectively. These matrices are used as design parameters to produce the desired performance.

The solution is given by [4]:

$$\underline{u}(t) = \mathbf{L}(t) \underline{x}(t)$$

where \underline{x} is the state vector at time t . The matrix $\mathbf{L}(t)$ is determined by:

$$\mathbf{L}(t) = \mathbf{V}^{-1}(t)\mathbf{B}^T\mathbf{S}(t) \quad (5.2)$$

where \mathbf{B} is a matrix mapping the control to the state. The matrix $\mathbf{S}(t)$ can be solved using a Ricatti equation:

$$\begin{aligned} \dot{\mathbf{S}}(t) &= \mathbf{S}(t)\mathbf{B}\mathbf{V}^{-1}(t)\mathbf{B}^T\mathbf{S}(t) - \mathbf{S}(t)\mathbf{A}(t) - \mathbf{A}^T(t)\mathbf{S}(t) + \mathbf{W}(t) \\ \mathbf{S}(t_f) &= \mathbf{S}_f \end{aligned}$$

It is worth noting that the Riccati equation can be solved by backwards propagation as a discrete process. Letting t_{i+1} correspond to the final epoch, and t_i be the epoch just previous, the recursion relationship can be written as [8]:

$$\begin{aligned} \mathbf{L}_i &= - [\mathbf{V}_d(t_{i+1}, t_i) + [\mathbf{B}_d(t_{i+1}, t_i)]^T \mathbf{S}_{i+1} \mathbf{B}_d(t_{i+1}, t_i)]^{-1} \times \\ &\quad [\mathbf{B}_d(t_{i+1}, t_i)]^T \mathbf{S}_{i+1} \mathbf{\Phi}(t_{i+1}, t_i) \\ \mathbf{S}_i &= [\mathbf{\Phi}(t_{i+1}, t_i)]^T \mathbf{S}_{i+1} \mathbf{\Phi}(t_{i+1}, t_i) - \\ &\quad (\mathbf{L}_i)^T [\mathbf{V}_d(t_{i+1}, t_i) + [\mathbf{B}_d(t_{i+1}, t_i)]^T \mathbf{S}_{i+1} \mathbf{B}_d(t_{i+1}, t_i)] \mathbf{L}_i + \mathbf{W}_d(t_{i+1}, t_i) \end{aligned}$$

If $u(t)$ is held constant over the interval $t - t_i$ then:

$$\begin{aligned} \mathbf{B}_d(t, t_i) &= \int_{t_i}^t \mathbf{\Phi}(t, \tau) \mathbf{B}(\tau) d\tau \\ \mathbf{W}_d(t_{i+1}, t_i) &= \int_{t_i}^{t_{i+1}} \mathbf{\Phi}(t, t_i)^T \mathbf{W}(t) \mathbf{\Phi}(t, t_i) d\tau \\ \mathbf{V}_d(t_{i+1}, t_i) &= \int_{t_i}^{t_{i+1}} [\mathbf{B}_d(t, t_i)^T \mathbf{W}(t) \mathbf{B}_d(t, t_i) + \mathbf{V}(t)] d\tau \end{aligned}$$

The matrix $\mathbf{\Phi}(t_{i+1}, t_i)$ is the state transition matrix, and its generation is discussed in Section 4.4.2.

5.2 Controller Implementation

Since the type of controller was not critical to the overall simulation, a simple tracking controller was implemented. In this scenario, the controller's job is to maintain a constant distance between the node and the hub satellite. The controller makes use of the absolute position and velocity estimates, and returns a maneuver to the node. The algorithm for the generation of the control vector is:

1. The absolute position of both the hub and the node spacecraft are propagated 30 seconds into the future. This is the epoch at which the control will be applied, and will be designated t_c . The 30 second buffer window allows ample time for the controller to run and the control vector to be distributed to the Remote Control Machine (defined in section 6.1.4).
2. The node spacecraft constructs the absolute state of the hub by differencing its own absolute state estimate and the relative state estimate. The reconstructed hub's absolute state is then used to generate a UVW reference frame centered at the hub spacecraft.
3. The desired relative state is constructed as an offset in this UVW frame. In this work, a 100 km separation was added in the \hat{v} direction to form the desired state.
4. The node's estimate of the hub position is then used to rotate the desired state back into the inertial reference frame. It should be noted that each node will have a slightly different estimate of the hub's position, and therefore a different UVW frame. The result of this offset is that the desired trajectory for each node will vary slightly from the ideal 100 km separation.

5. The desired state is propagated forward to some epoch in the future, t_f , at which \mathbf{S}_f is defined. An arbitrary coasting period of five minutes was chosen for this controller. Accordingly, the desired state is integrated 5 minutes into the future, and the Ricatti equation is back propagated 5 minutes as well. At the end of the coasting window, a new control is calculated and the process continues.
6. The Ricatti Equation is solved by back propagating from the epoch t_f to the control epoch t_c in one second increments. The state transition matrix used in this procedure is generated using the same plant formulation that the filter used.
7. The back propagation yields the $\mathbf{L}(t_c)$ matrix, which is used to find the control vector at time t_c .

The algorithm presented was selected arbitrarily, and may not be efficient enough for actual flight hardware. One option to conserve computational time would be to generate an entire orbit worth of gains and store them in RAM. This would alleviate having to perform the back propagation every 5 minutes.

5.3 Offline Controller Tuning

An empirical approach was selected to determine the control parameters that would optimize the linear controller. A “cheap” controller was used to simplify the search space for the desired values. In this situation, the terminal penalty matrix \mathbf{S}_f was set to zero. The state penalty matrix was set to an identity matrix, and the control penalty matrix was constructed as the

appropriately sized identity matrix premultiplied by the scalar parameter α . This scalar was adjusted until the desired results were obtained.

The controller was tested offline by providing it with simulated absolute state estimates. In order to verify that the controller could track the desired position without error, the input states were noise free. The node satellite was offset from the desired initial state according to the values given in Table 5.2. The simulation spanned one orbit, and a control vector was output every 300 seconds. The maneuvers were assumed to be ideal, and always along the inertial axes.

Figure 5.1 depicts a case where the control penalty matrix α has a value of $5e6$. This value is too high and does not allow enough actuation effort to drive the tracking error to zero. The tracking error is defined as the difference between the estimated state and the desired state. With an α value of $5e5$, there is too little penalty applied and the controller consistently overshoots the desired trajectory. This can be seen in Figure 5.2. With the desired control solution bracketed, the α value was adjusted slightly until the results shown in Figure 5.3 were achieved. The design parameters are summarized in Table 5.1.

Table 5.1: Controller Design Parameters

Parameter	Value
S_f	0
W	0
V	$1e6$

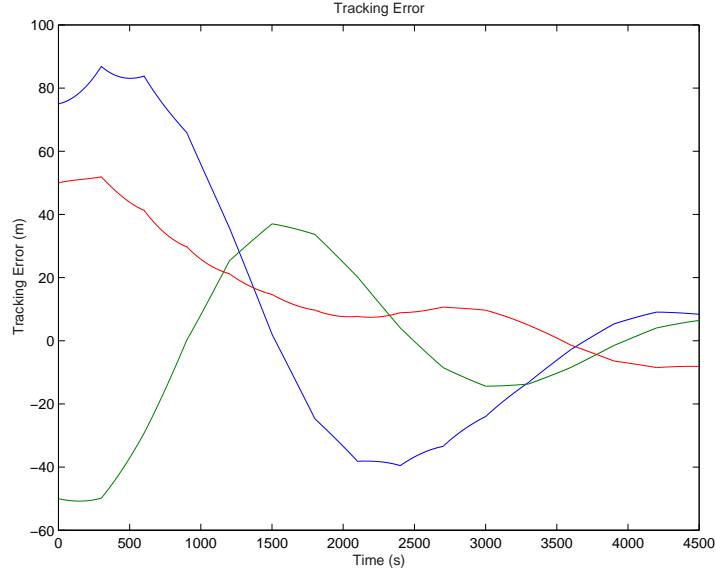


Figure 5.1: Tracking Error With Too Much Control Penalty.

The controller targets a desired absolute position based upon the propagated absolute state estimates. A typical pseudorange-based filtered GPS solution contains approximately two to three meters of error per axis. Therefore, to study the predicted controller performance, the simulated absolute states in this offline test were augmented with two meters of white noise before being sent to the controller. Figure 5.4 demonstrates that the controller has a 20 meter limit on its tracking accuracy. It should be emphasized that this is due to the relatively low accuracy of the absolute states sent to the controller. If more emphasis was placed upon improving this accuracy, the overall tracking error would decrease accordingly. This fact is verified in Figure 5.5, where the estimated state noise is approximately one-half meter and the tracking error drops to within a few meters.

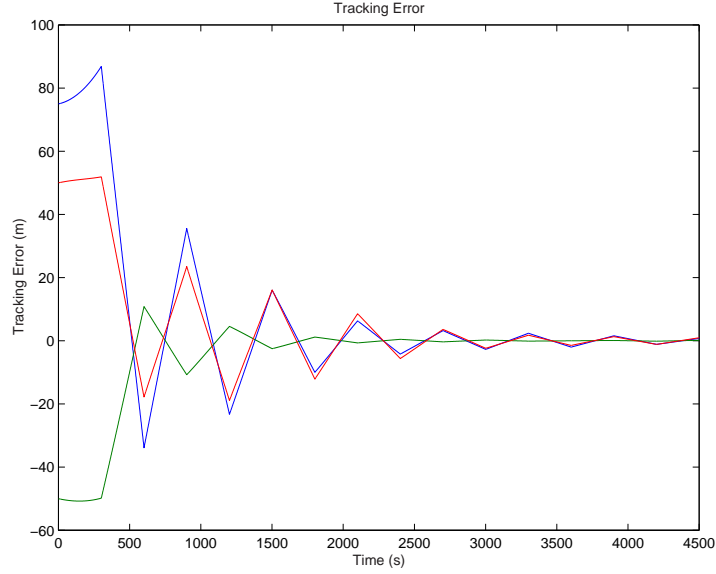


Figure 5.2: Tracking Error With Too Little Control Penalty.

As a final verification step, the maneuver window was decreased to ten seconds, and the noise on the absolute state was returned to two meters. With a ten second control window, the errors induced by lengthy propagation of inaccurate states are minimized, and the controller is able to track the desired trajectory to within the two meter input noise. Figure 5.6 demonstrates this tracking capability, and verifies that no stray tracking errors are leaking into the solution.

5.4 Formation Control Architecture

Just as there are many different types of controllers, there are several different ways to implement a control architecture for a formation. Three such architectures are the centralized controller, decentralized controller, and a hy-

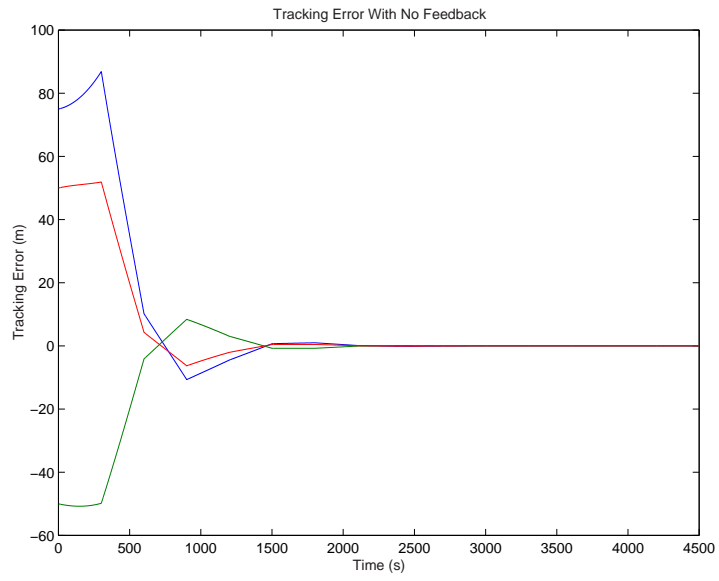


Figure 5.3: Tracking Error With Appropriate Penalty.

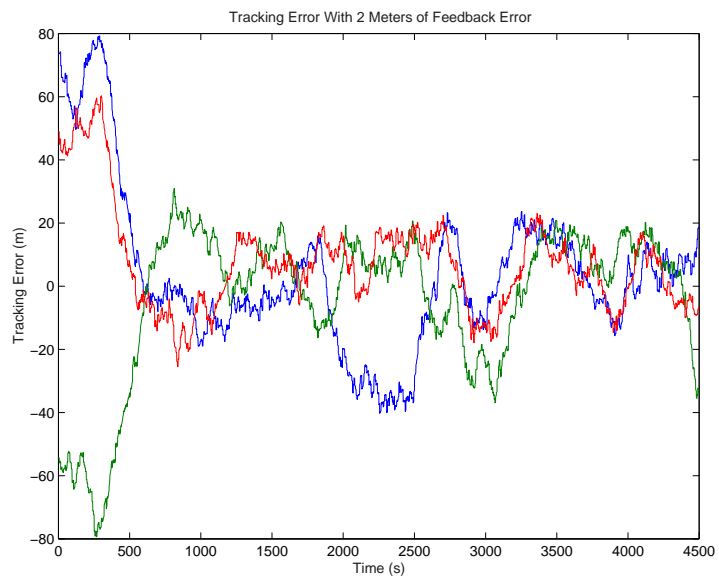


Figure 5.4: Tracking With 2 m Of Measurement Noise at 300 Sec.



Figure 5.5: Tracking With .5 m Of Measurement Noise at 300 Sec.

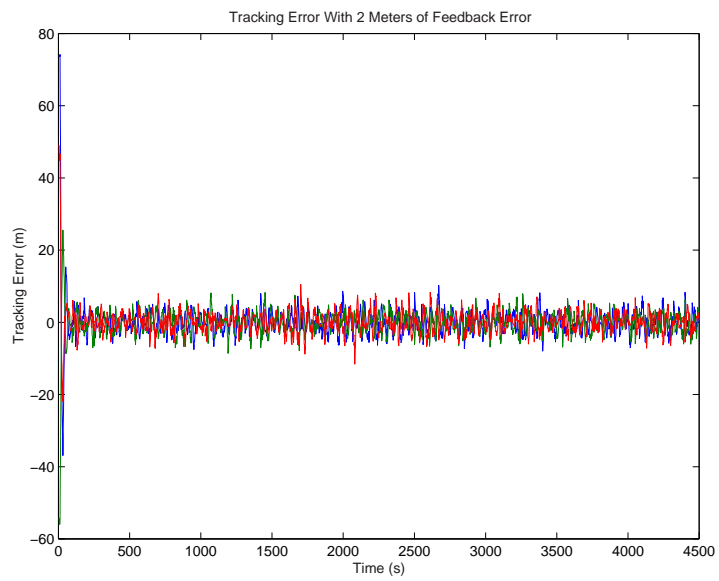


Figure 5.6: Tracking With 2 m Of Feedback Noise at 10 Sec.

Table 5.2: Initial Node Absolute State Offset From The Desired Trajectory

State	Deviation
u	75 m
v	-50 m
w	50 m
\dot{u}	.01 m/s
\dot{v}	-.05 m/s
\dot{w}	.2 m/s

brid controller that leverages the strengths of the centralized and decentralized controllers[7]. These architectures are illustrated in Figure 5.7.

A form of decentralized architecture was implemented for this work, and is depicted in Figure 5.8. The simulation environment was designed to allow for rapid adaptation to any of these architecture types, and would make an excellent tool to study the benefits and drawbacks of each technique.

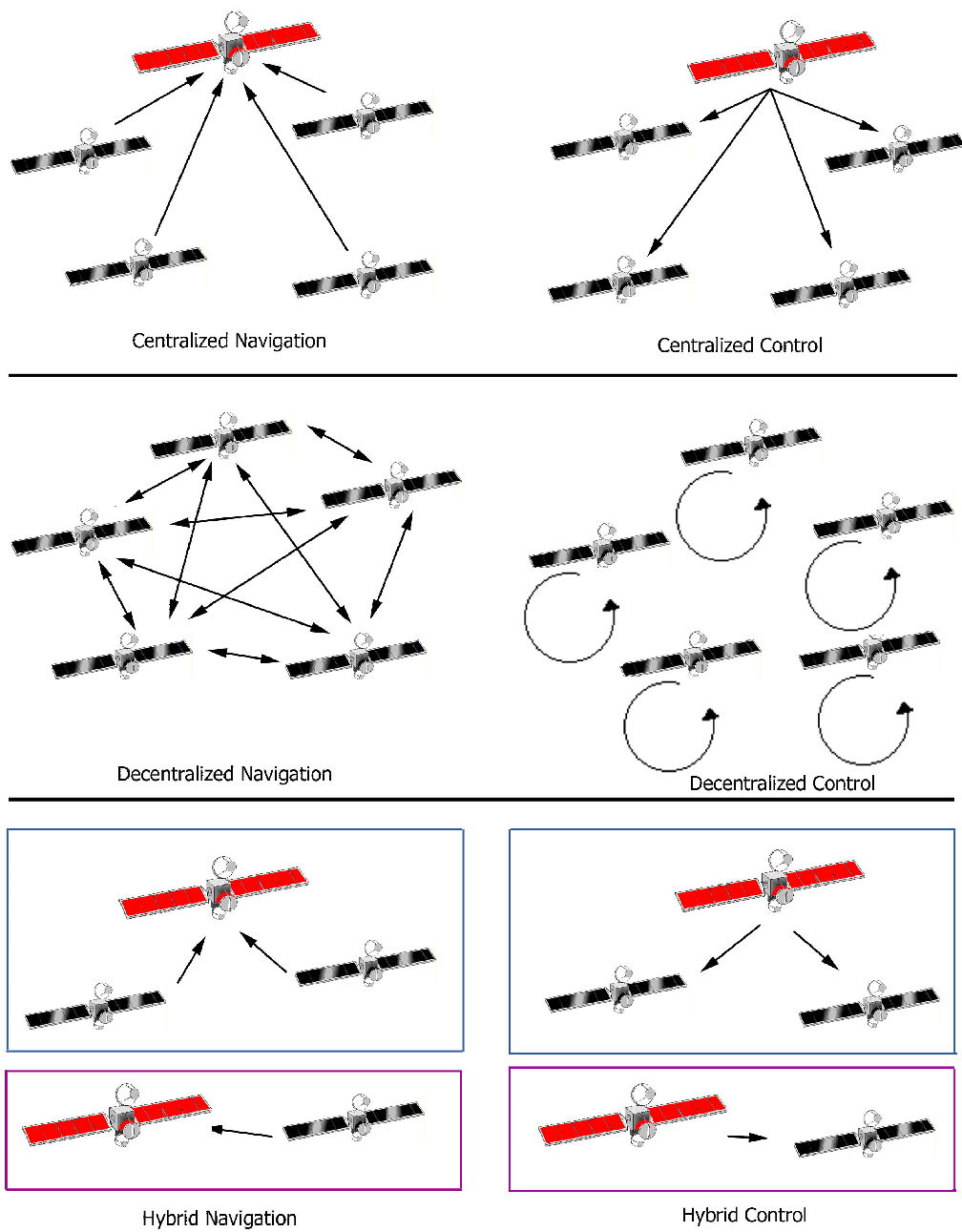


Figure 5.7: Control Architectures

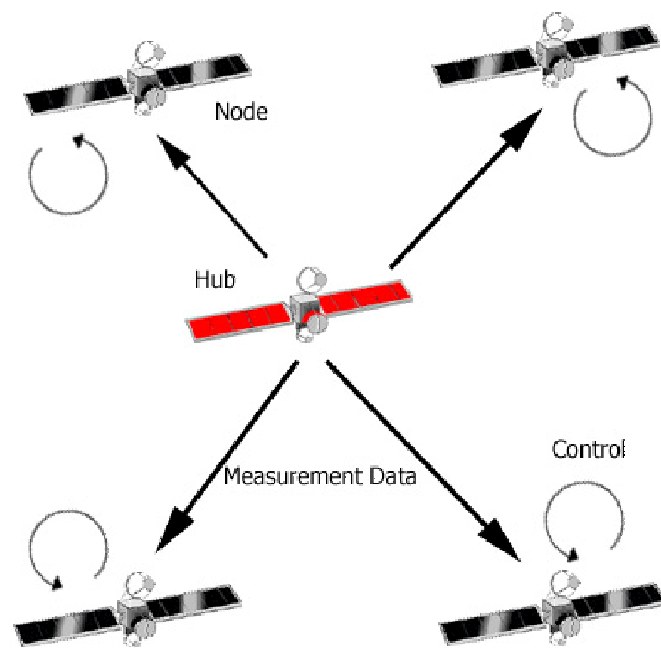


Figure 5.8: Implemented Decentralized Architecture

Chapter 6

Extended Formation Flying Testbed

The EFFTb is a combination of hardware and software that allows for the simulation of multiple vehicles in a real-time closed-loop hardware-in-the-loop situation. Each of the major components is described in the following subsections.

6.1 Hardware Description

The EFFTb hardware was customized for a Low Earth Orbit (LEO) using GPS navigation signals. The design of the testbed was generalized to allow for multiple measurement sources and dynamic inputs. For example, to accommodate a deep space formation, the GPS simulator/receiver measurement source could be replaced with a simulated star tracker, and the dynamic modelling software could be replaced with the appropriate equations of motion. In this way, the EFFTb can be used to simulate nearly any proposed formation. The following descriptions detail the hardware as configured for this work. Figure 6.1 illustrates this hardware design.

6.1.1 GPS Signal Simulator

The heart of the simulation setup is the GPS signal simulator. Each of the participating installations is required to have at least one such simulator

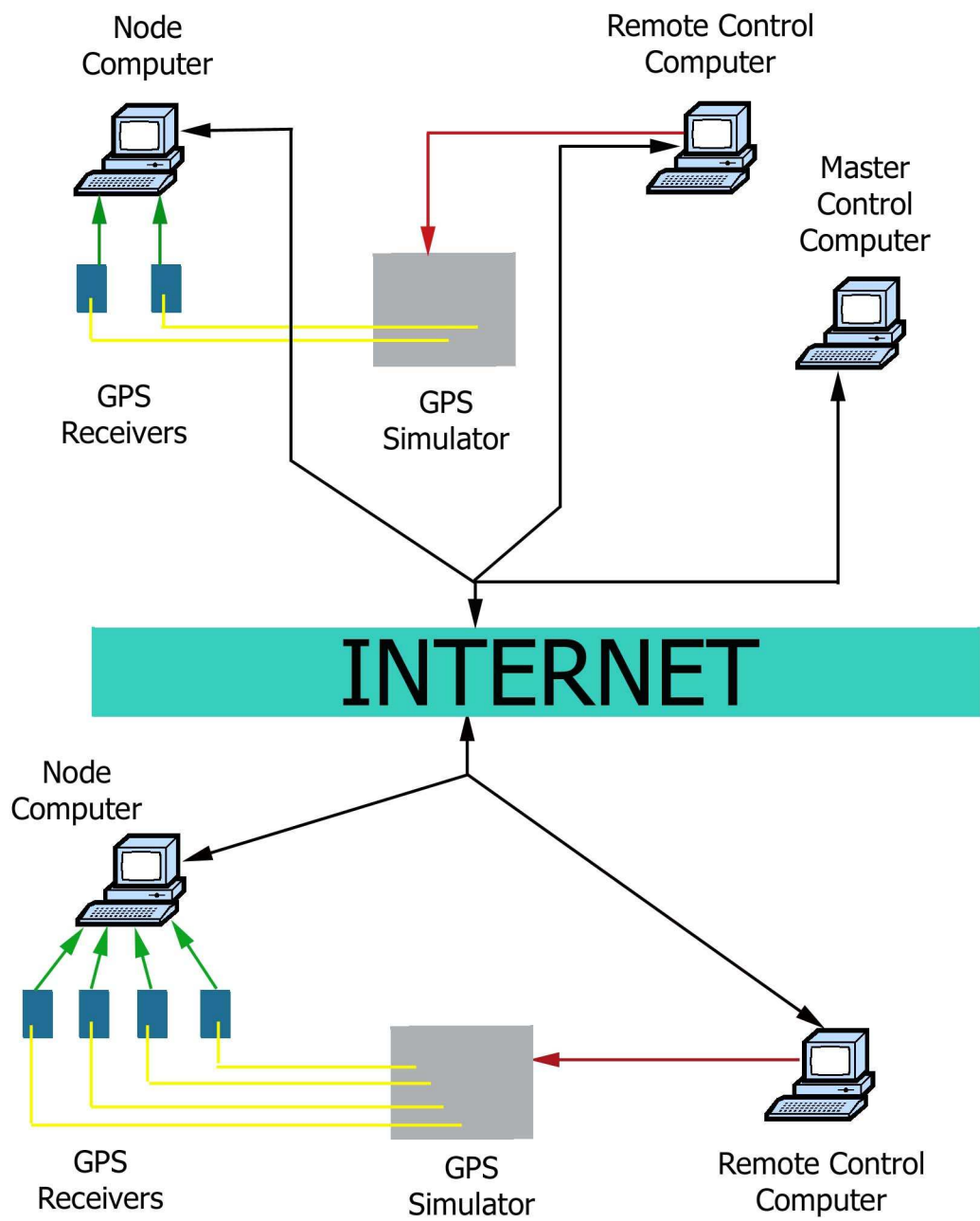


Figure 6.1: Hardware Setup.

onsite. The GPS laboratory at the University of Texas at Austin/Center for Space Research (UTCSR) uses a Spirent STR4760 GPS signal simulator. This simulator is capable of simulating L1 signals for two vehicles on up to 16 channels each. All of the environmental variables affecting the GPS signals, such as ionospheric delay and GPS clock errors, are controlled by the simulator. The main goal of the EFTB is to overcome the limitations of two RF outputs on this simulator by connecting several such simulators together.

6.1.2 Orion GPS Receivers

The Orion type GPS receiver was chosen to provide measurements. The Orion receiver is based on the terrestrial GPS receiver built around the Zarlink GP2000 chipset [1]. The Orion receiver is capable of tracking 12 satellites on



Figure 6.2: Orion Receivers.

the L1 frequency. In order to optimize the receiver performance for space use, several modifications were made to the receiver [21]. These modifications include the fixes related to the implicit assumption of a low speed vehicle in the Doppler prediction and the correction of the time tagging error [11]. In order to facilitate measurement exchange between receivers, a clock steering algorithm has been implemented in the receiver which aligns the measurement epochs

and the navigation solutions to the integer GPS second. The Orion also utilizes a 3rd order phase lock loop (PLL) assisted by a 2nd order frequency lock loop (FLL) to improve the accuracy of the integrated carrier phase measurement.

6.1.3 Master Control Computer

The Master Control Computer (MCC) functions as the nerve center of the EFFTb. As denoted in 6.3 this is the only machine that doesn't have to be collocated with the simulation equipment. The functions of the MCC are described below:

1. Formation Initialization

The MCC possesses the initialization data for all of the GPS receivers and the Remote Control Machines. When the simulation is initialized, the MCC sends out the appropriate two-line element set to the each node. The ECEF position of each node as well as its mass and area characteristics are then sent to the Remote Control Computer for propagation.

2. Simulation Activation

In order to keep the dynamic environments at all facilities within a second of each other, an electronic activation has been designed. Once all of the initialization is completed, the MCC sends out a pulse alerting each machine that the simulation has begun. Upon receiving this pulse, the Node Computers upload the two-line elements into their respective receivers, and the Remote Control Computers begin to propagate the equations of motion and feed the simulator the trajectory data. Accordingly, the difference between the start times at each facility is simply the

signal transmit time through the internet.

3. Data Relay

In order to emulate a true broadcast environment via hardwired cables, a mock communication system has been developed. In this simulation, no node has the ability to communicate directly with any other node. All packets are instead sent to the MCC, which then rebroadcasts them to all of the nodes. The communication system on each node must be robust enough to recognize the information sent to it, identify the relevant data and reject the rest. The MCC also routes the control information directly to the Remote Control Computer.

4. Data Logging

The MCC computer allows for a convenient central point for data logging and archiving. This requires extra effort on the part of the node to distribute the position and covariance information for the state, but it relieves the user of the inconvenience of collecting multiple files on several different systems.

6.1.4 Remote Control Computer

The Remote Control Computer (RCC) is responsible for generating the dynamics for the simulation. While the Spirent simulator is capable of performing this task, the closed-loop environment requires an external data source. The RCC is initialized with each of the node's initial positions, and propagates them forward at 10 Hz according to the user defined environment variables. At the appropriate time, the RCC augments the node's velocity vector with the Δv provided by the Node Computer. The RCC smooths this

impulse by integrating it over one second. Ten times each second the RCC sends the trajectory information to the simulator, which then generates the appropriate GPS signals. The RCC at each location runs independently of the remainder of the RCC's. As long as the filter update intervals are sufficiently large, strict time synchronization of these machines is not necessary. However, if a rapid update rate is required, the RCC or the simulators may need to be driven by a precise clock in order to maintain system wide synchronization.

6.1.5 Hub Computer

The satellite that is designated the hub is only required to broadcast its measurements at each epoch. The Hub Computer (HC) extracts the pseudorange, doppler and carrier phase measurements from the receiver, as well as the position fix solution, and forwards the data on to the MCC. The MCC then distributes the measurements to the formation

6.1.6 Node Computer

The Node Computer (NC) performs all of the estimation and control computations. One NC can administer multiple GPS receivers, each utilizing its own instance of the node software. When the NC communication routine detects measurements with identical time-tags, it invokes the estimator to determine the current state. If it is time to perform a station keeping maneuver, the node calculates this maneuver and sends it to the RC.

6.2 Software Description

The EFFTb code was developed in standard C. The primary utility is the generality of the code structure. The NC software was developed with the estimator and controller being subroutines called by the main shell. The subroutines can easily be swapped out with other, more sophisticated programs. The RC requires only a control vector, and is impartial to how it is generated. The RC software can easily be modified to write to a device other than the GPS signal generator. The overall flow of the software is illustrated in 6.3, where the dashed arrows represent data transfer.

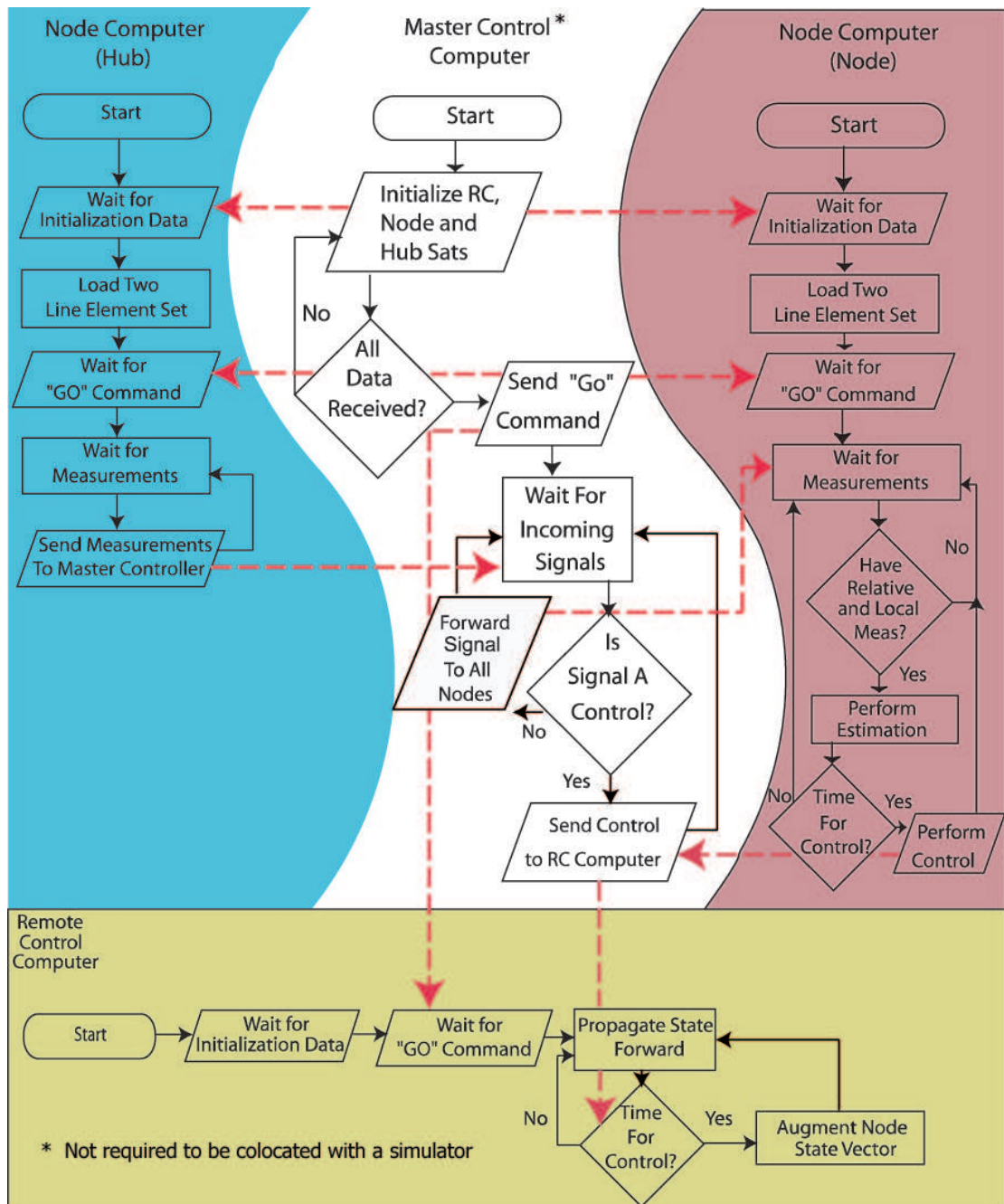


Figure 6.3: Software Flowchart.

Chapter 7

Results

This chapter details the results obtained from the simulation environment for three satellite separation distances. These tests include runs made in offline simulations as well as the filter and controller performances when inserted into the EFFTb. Tuning parameters for both the observer and controller are also documented.

7.1 Offline Filter Validation

The Kalman filter was tuned using measurement data from the GPS receivers and processed in an offline environment. In these tests only two satellites were used, the node satellite and the hub satellite. The simulator settings for the “truth” environment are listed in Table 7.1. Since the goal was to focus on how the intersatellite distances affect the navigation error, the GPS clock and ephemerides errors in the GPS simulator were disabled. To evaluate the filter’s performance over various distances, three candidate distances were chosen. The satellites were placed in the same orbit plane with an inclination of 87 degrees and altitude of approximately 600 km. Spacing of the nodes was accomplished by adjusting the true anomaly. The true ECI initial conditions for each of these cases are provided in Table 7.2.

The initial error covariance matrix is diagonal, and the initialization

Table 7.1: Simulator Environment Settings.

Parameter	Setting
Earth Gravity Model	10x10
Cross Sectional Area	2 m^2
C_D	2
Mass	100 Kg
GPS Clock Error	Off
GPS Ephemerides Errors	Off
Ionosphere	Constant Thickness with Sinusoidal Variation

Table 7.2: True Initial Conditions: November 6 2001 00:00:00 GPS.

	Hub	Node 1 km	Node 100 km	Node 500 km
X m	-4.81976e6	-4.81980e6	-4.82294e6	-4.82531e6
Y	4.81976e6	4.81972e6	4.81555e6	4.78837e6
Z	0.00000	0.00099e6	0.09975e6	0.49837e6
Vx m/s	-0.28313e3	-0.28234e3	-0.20390e3	0.11332e3
Vy	-0.28313e3	-0.28393e3	-0.36231e3	-0.67808e3
Vz	7.64046e3	7.64046e3	7.63964e3	7.61998e3

parameters are listed in Table 7.3.

The error covariance terms for the GPS ephemerides errors and the double-differenced integer ambiguities must be reset with the inclusion of each new GPS satellite. This is accomplished by resetting the corresponding diagonal element and zeroing out the associated row and column elements in the error covariance matrix.

To examine the filter performance, and perform the necessary tuning,

Table 7.3: Error Covariance Initialization

State		Value
u, v, w	m	400
$\dot{u}, \dot{v}, \dot{w}$	m/s	1
$\Delta u, \Delta v, \Delta w$	m	400
$\Delta \dot{u}, \Delta \dot{v}, \Delta \dot{w}$	m/s	1
b	m/s	10
\dot{b}	m/s^2	1
$\Delta \lambda N$	m	25
B	m	100

GPS data was analyzed offline. The data were generated using the internal propagator in the GPS simulator. Two Orion receivers were utilized to process the measurements, which were then saved to text files. These text files were used as input to the filter to generate the results in this section.

7.1.1 Filter Performance at 1 Hz

To determine the optimal performance, the measurement update rate was set at 1 Hz. Figures 7.1 and 7.2 show the absolute position and velocity errors for the node spacecraft. Since the absolute states are independent of the relative distances, these figures are not repeated for the longer baselines. It was observed that the accuracy of the absolute position is typical for a GPS pseudorange based navigation solution. The jumps in the error and error covariance are caused by the introduction and removal of satellites from the solution. The accuracies for the absolute states are listed in Table 7.4. The accuracy of the relative state is not strongly correlated with the accuracy of the absolute states. For short baselines, it was shown that the estimation of

Table 7.4: Absolute State Error in $\hat{u}, \hat{v}, \hat{w}$ frame (mean \pm std)

State		Accuracy
u	m	-0.0034 ± 3.4934
v		-0.0450 ± 3.0136
w		-0.0004 ± 2.4191
\dot{u}	m/s	0.0703 ± 0.0335
\dot{v}		-0.0185 ± 0.0267
\dot{w}		-0.0092 ± 0.0415

the absolute state could be replaced by simply propagating the absolute state forward with little lack of accuracy [28].

The relative accuracies were determined by generating an ensemble average over five simulation runs for each baseline length. The results are depicted in Figures 7.3 - 7.8. The filter parameters for each baseline are detailed in Table 7.5. It should be noted that the large process noise values for the clock bias and drift reflect the inability to accurately model these states. Table 7.6 details the steady-state accuracies for these simulations.

The errors at the one kilometer baseline are comparable to those found in current literature [29] [11]. The convergence for the relative position is dependent on the correct resolution of the double differenced integer ambiguity states. For the one second update rate, this convergence takes approximately 1000 seconds. For the 1 km and 100 km case, once these integers are determined, the solution is only slightly disturbed by the inclusion of new satellite

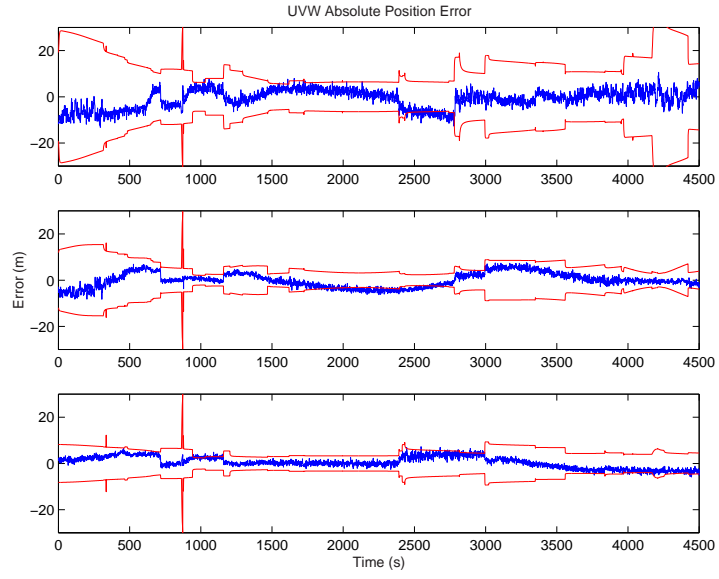


Figure 7.1: Ensemble Mean Absolute Position Actual and Formal Errors.

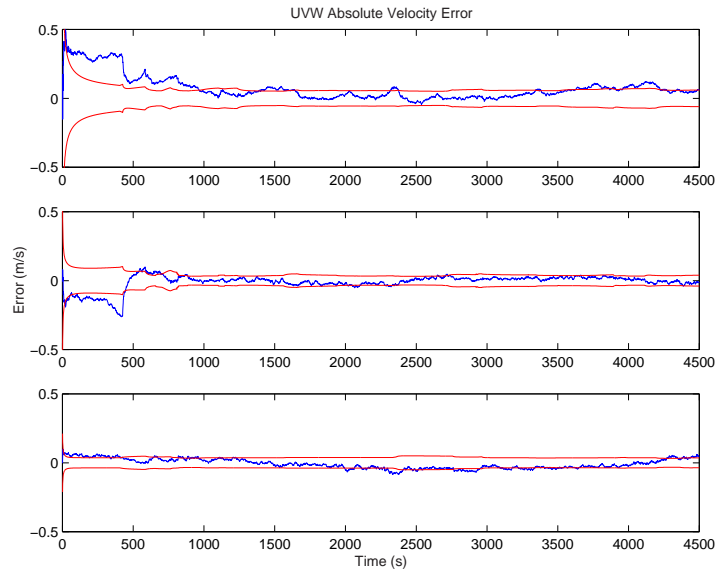


Figure 7.2: Ensemble Mean Absolute Velocity Actual and Formal Errors.

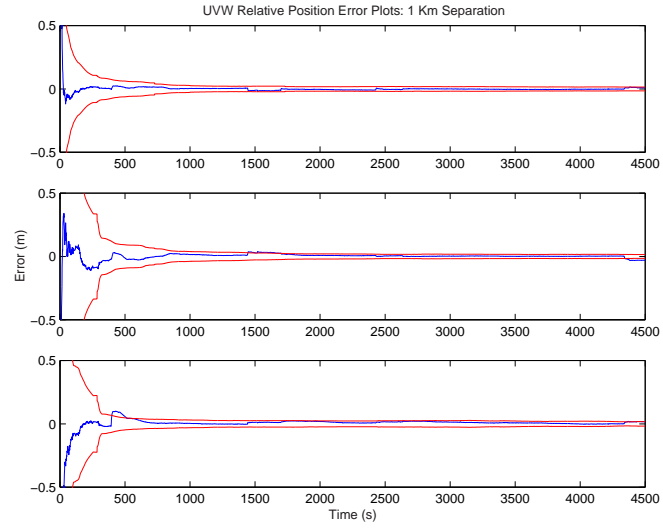


Figure 7.3: 1 km Ensemble Mean Relative Position Actual and Formal Errors/1 Second Update.

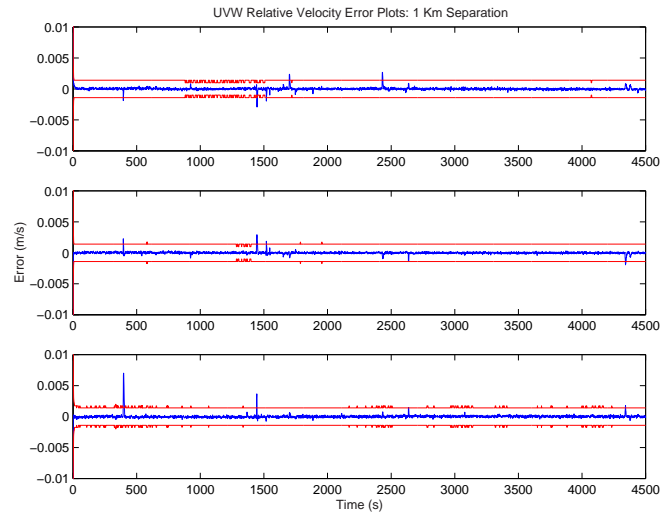


Figure 7.4: 1 km Ensemble Mean Relative Velocity Actual and Formal Errors/1 Second Update.

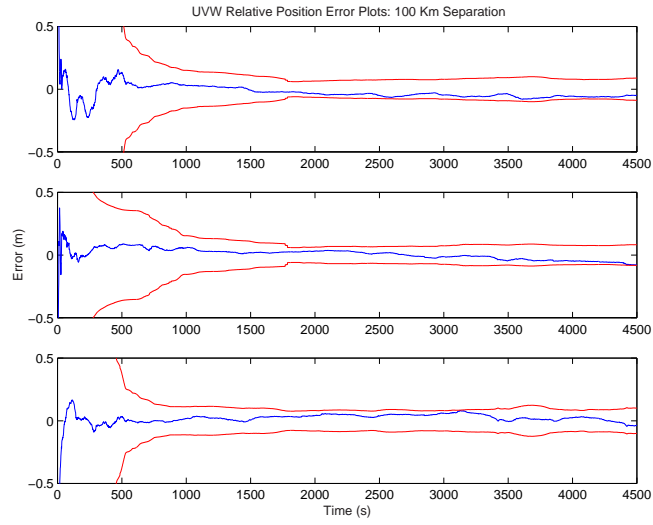


Figure 7.5: 100 km Ensemble Mean Relative Position Actual and Formal Errors/1 Second Update.

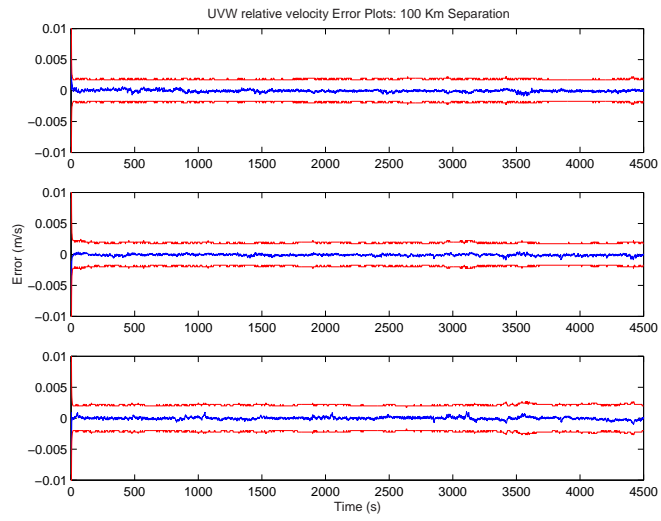


Figure 7.6: 100 km Ensemble Mean Relative Velocity Actual and Formal Errors/1 Second Update.

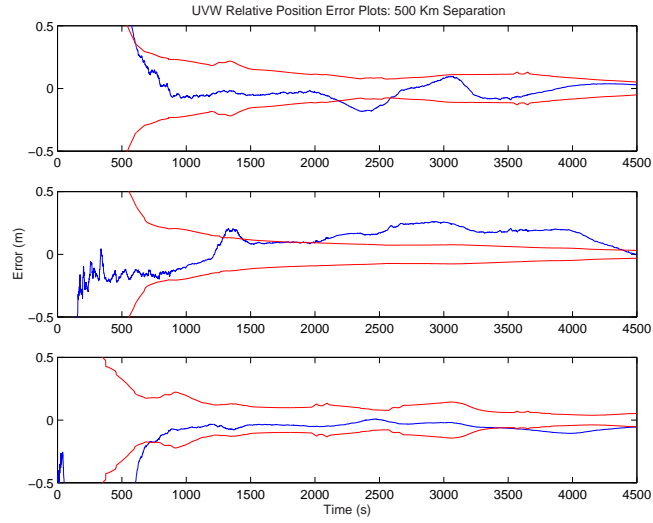


Figure 7.7: 500 km Ensemble Mean Relative Position Actual and Formal Errors/1 Second Update.

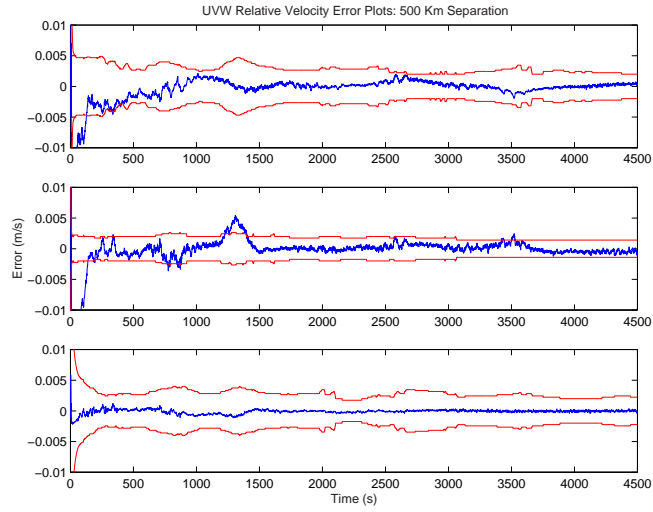


Figure 7.8: 500 km Ensemble Mean Relative Velocity Actual and Formal Errors/1 Second Update.

Table 7.5: Process Noise Settings: 1Hz Update

Parameter		1 km Case	100 km Case	500 km Case
$\sigma_{r_{ABS}}^2$	m^2	1e-2	1e-2	1e-1
$\sigma_{v_{ABS}}^2$	$\frac{m^2}{s^2}$	5e-5	5e-5	1e-5
$\sigma_{r_{REL}}^2$	m^2	1e-6	1e-6	1e-7
$\sigma_{v_{REL}}^2$	$\frac{m^2}{s^2}$	1e-7	1e-7	1e-6
σ_b^2	m^2	1e6	1e6	1e6
$\sigma_{\dot{b}}^2$	$\frac{m^2}{s^2}$	1e6	1e6	1e6
$\sigma_{\lambda N}^2$	m^2	1e-6	1e-5	1e-5
σ_B^2	m^2	1e-5	1e-5	1e-5

Table 7.6: Relative State Error: 1 Sec. Update Rate (mean \pm std)

State		1 km Case	100 km Case	500 km Case
Δu	m	-0.0024 ± 0.0044	-0.0478 ± 0.0149	-0.0307 ± 0.0644
Δv		0.0041 ± 0.0105	-0.0114 ± 0.0324	0.0989 ± 0.0627
Δw		0.012 ± 0.0063	0.0310 ± 0.02376	-0.0501 ± 0.0281
$\Delta \dot{u}$	m/s	0.00001 ± 0.0002	-0.0000 ± 0.0001	0.0001 ± 0.0005
$\Delta \dot{v}$		-0.00001 ± 0.0002	-0.0001 ± 0.0002	0.0000 ± 0.0005
$\Delta \dot{w}$		0.00002 ± 0.0001	0.0000 ± 0.0002	-0.0000 ± 0.0001

pairs.

The 500 km ensemble does not converge as well as the other two scenarios. In two of the simulations, the filter struggled to accurately estimate the ambiguities in the face of the greater differential ionosphere delay error. The ensemble averaging reflects this difficulty in larger errors. Another factor limiting the accuracy at the long baseline is the number of common view GPS satellites available. The number of satellites utilized in the double difference is illustrated in Figure 7.9. While the 500 km baseline has five or more satellites

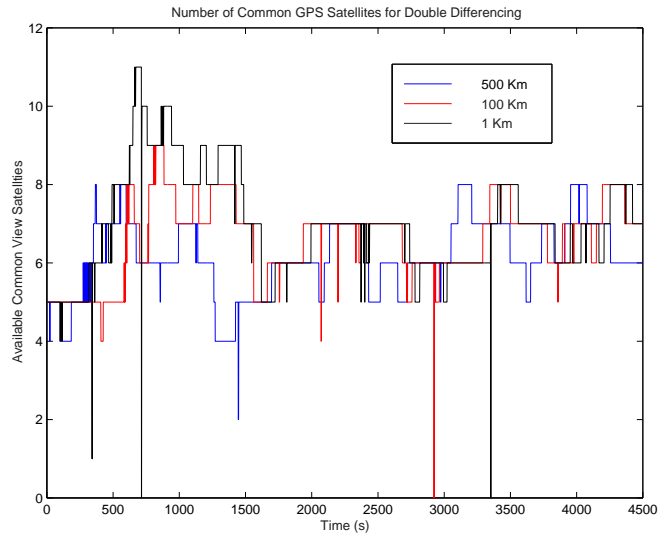


Figure 7.9: Common View GPS Satellites.

in view for a majority of the time, there is a ten minute time period where there are four or fewer satellites available. It is apparent that the filter as constructed has a maximum relative baseline length near 500 km.

7.1.2 Filter Performance at 5 Seconds

In an actual distributed system, a one second update rate may pose too heavy of a burden on the computer and communication systems. Accordingly, the measurement update rate of the filter was slowed to 5 seconds. The filter gains were adjusted to account for the larger emphasis placed upon the dynamical models. The process noise covariance values are listed in 7.7. The ensemble results for the five second update rate are displayed in 7.10-7.15.

Table 7.7: Process Noise Settings: 0.2 Hz Update				
Parameter		1 km Case	100 km Case	500 km Case
$\sigma_{r_{ABS}}^2$	m^2	1e-2	1e-1	1e-1
$\sigma_{v_{ABS}}^2$	$\frac{m^2}{s^2}$	5e-4	5e-4	1e-4
$\sigma_{r_{REL}}^2$	m^2	1e-5	1e-4	1e-4
$\sigma_{v_{REL}}^2$	$\frac{m^2}{s^2}$	1e-5	1e-6	1e-5
σ_b^2	m^2	1e6	1e6	1e6
σ_i^2	$\frac{m^2}{s^2}$	1e6	1e6	1e6
$\sigma_{\lambda N}^2$	m^2	1e-6	1e-5	1e-4
σ_B^2	m^2	1e-5	1e-5	1e-5

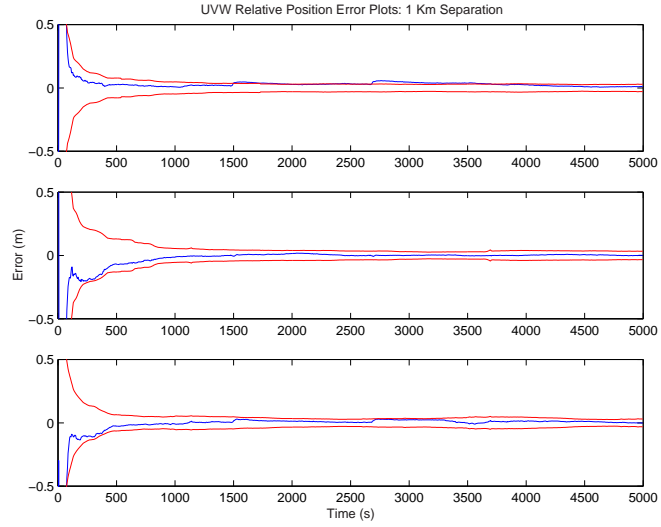


Figure 7.10: 1 km Ensemble Mean Relative Position Actual and Formal Errors/5 Second Update.

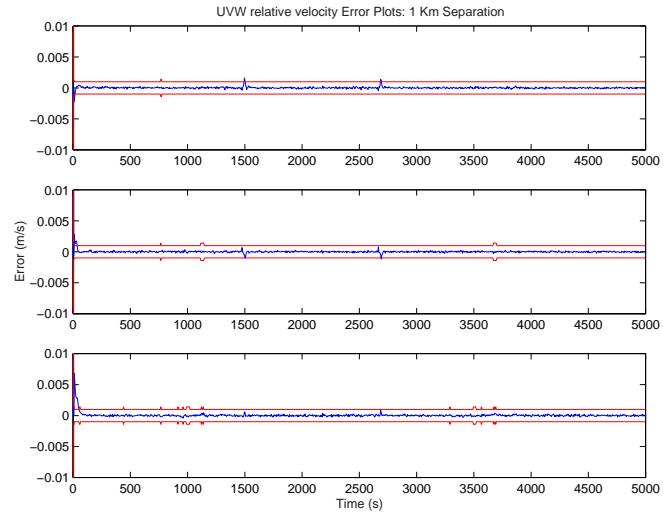


Figure 7.11: 1 km Ensemble MeanRelative Velocity Actual and Formal Errors/5 Second Update.

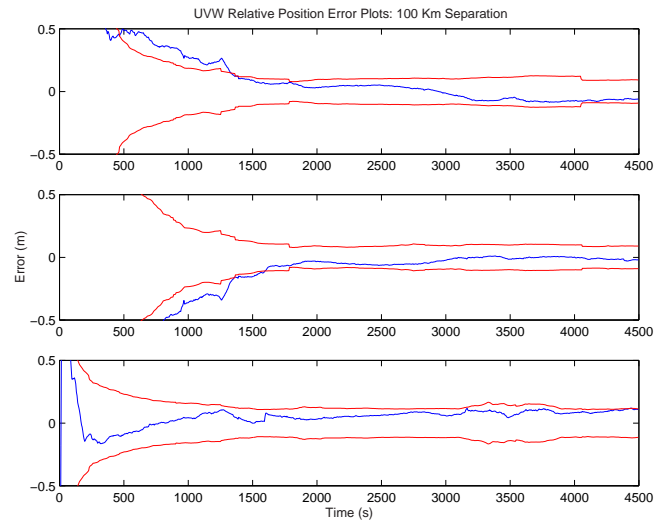


Figure 7.12: 100 km Ensemble Mean Relative Position Actual and Formal Errors/5 Second Update.

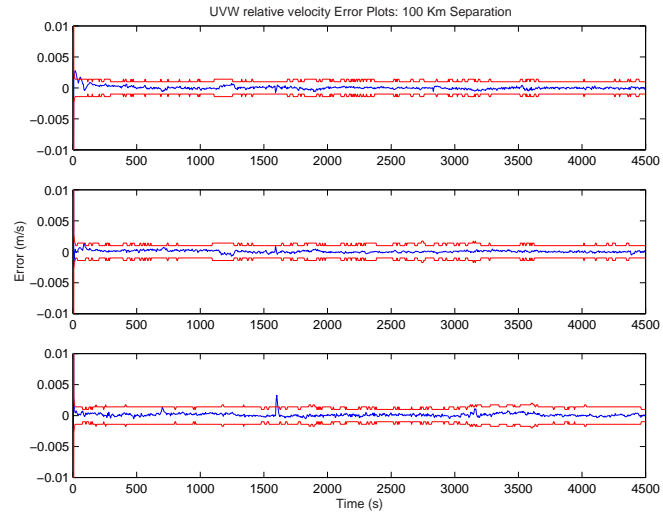


Figure 7.13: 100 km Ensemble Mean Relative Velocity Actual and Formal Errors/5 Second Update.

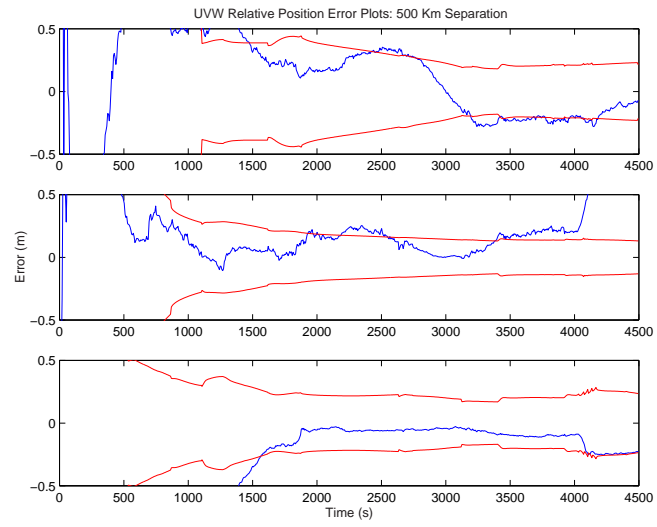


Figure 7.14: 500 km Ensemble Mean Relative Position Actual and Formal Errors/5 Second Update.

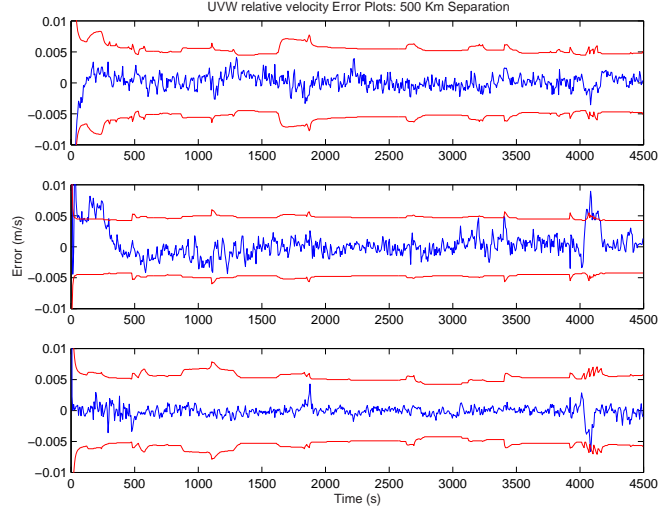


Figure 7.15: 500 km Ensemble Mean Relative Velocity Actual and Formal Errors/5 Second Update.

Table 7.8: Relative State Error: 5 Sec. Update Rate (mean \pm std)				
State		1 km Case	100 km Case	500 km Case
Δu	m	0.0295 ± 0.0129	-0.0116 ± 0.0566	-0.0127 ± 0.2237
Δv		0.0035 ± 0.0048	-0.0310 ± 0.0317	0.2308 ± 0.2141
Δw		0.0122 ± 0.0001	0.0718 ± 0.0252	0.2308 ± 0.0676
$\Delta \dot{u}$	m/s	0.0000 ± 0.0001	-0.0000 ± 0.0001	0.0000 ± 0.0016
$\Delta \dot{v}$		-0.00001 ± 0.0002	0.0001 ± 0.0000	0.0004 ± 0.0005
$\Delta \dot{w}$		0.00000 ± 0.0001	0.0001 ± 0.0003	-0.0001 ± 0.0008

Decreasing the update rate had little impact on the overall accuracies of 1 km and 100 km simulations. The steady state position noise levels are still better than 2 cm and 6 cm respectively. The inability of the filter to correctly resolve the double differenced integer ambiguities became more pronounced for the 500 km simulation. The large jump around 4000 seconds occurred in one of the runs when the number of common view satellites dropped to three, causing the relative position estimate to jump dramatically.

7.1.3 Cycle Slips

The integer ambiguity, as described in Chapter 3, is the number of whole carrier waves between the GPS satellite and the receiver. As long as the receiver keeps locked on the carrier, this integer is constant. If the receiver momentarily loses the carrier signal, the ambiguity may change by one or more whole cycles, resulting in a cycle slip. This slip would then have to be detected and accounted for in the filter in order to correctly process the carrier-phase measurement.

The Orion receivers manufactured at UT/CSR are prone to incur cycle slips, possibly due to the hardware fabrication process, or perhaps the hardware or software itself. These jumps usually manifest themselves in the form of a half-cycle slip. Since the carrier signal has a 19 cm wavelength, a half-cycle slip causes a 10 cm jump in the observed measurement. These slips are shown in Figure 7.24. Each spike occurring at a multiple near 10 cm indicates a cycle slip. Longer spikes indicate a poorly initialized ambiguity, which was reset during the next measurement update.

In order to minimize the impact of these slips, the filter checks the residuals for outliers at each measurement update. If a residual is greater than

9 cm, it is excluded from the solution. If this satellite combination generates two successive outliers, the filter assumes that a cycle slip has occurred. On the next measurement update, the ambiguity term and the corresponding error covariance element are reinitialized. The drawback to this method occurs when the one reference GPS satellite, the satellite against which all of the others are differenced, experiences a cycle slip. That error then saturates all of the double-differenced carrier phase measurements. When passing through the filter, all of the phase measurements are considered outliers, and the filter is left with no carrier-phase measurements for two time steps. A double-differencing algorithm which computed the differences successively would rectify this design weakness.

7.1.4 Best Case Filter Performance

In order to establish the best case filter performance, the two most stable Orion receivers were used, minimizing the impact of cycle slips. Figures 7.16-7.23 show these single run simulations for both the one second and the five second update rate. The accuracy for each scenario is listed in Table 7.9.

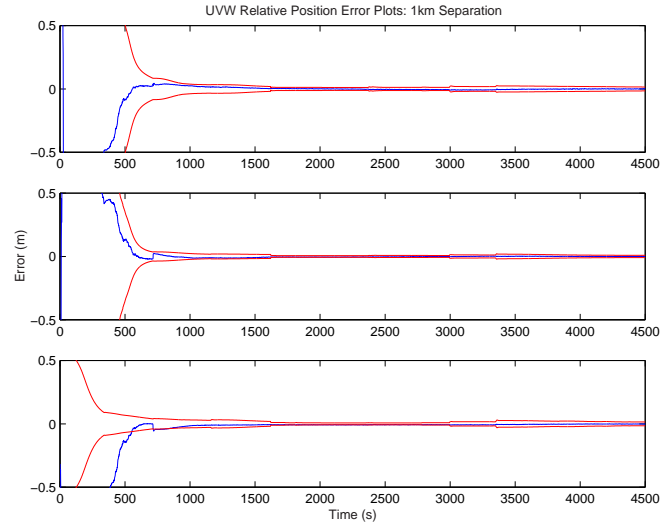


Figure 7.16: Best Case 1 km Relative Position Actual and Formal Errors/1 Second Update.

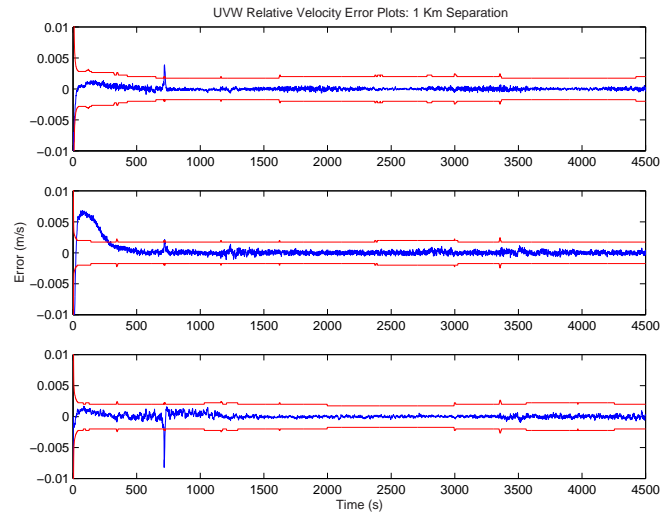


Figure 7.17: Best Case 1 km Relative Velocity Actual and Formal Errors/1 Second Update.

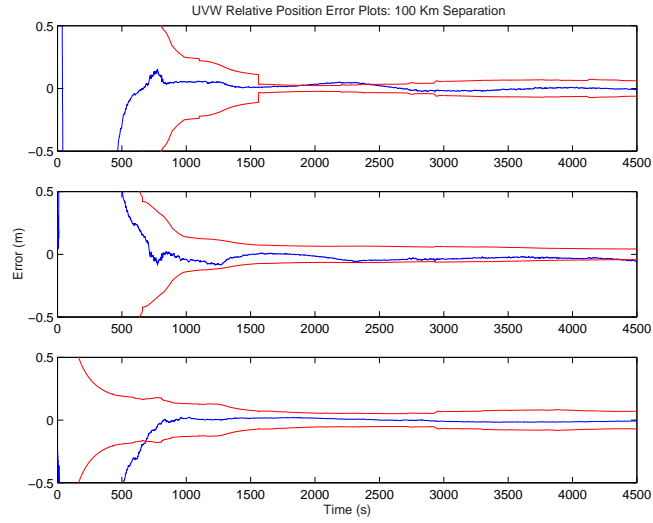


Figure 7.18: Best Case 100 km Relative Position Actual and Formal Errors/1 Second Update.

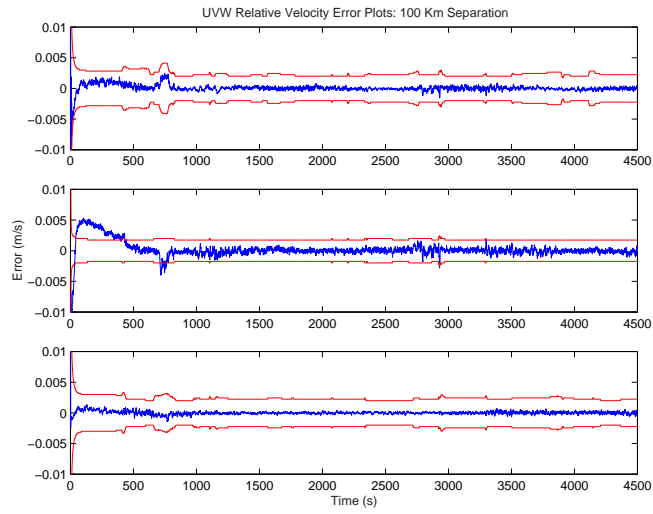


Figure 7.19: Best Case 100 km Relative Velocity Actual and Formal Errors/1 Second Update.

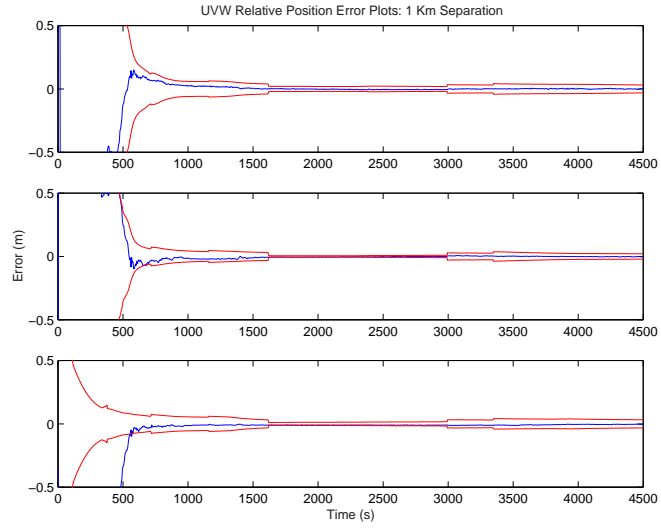


Figure 7.20: Best Case 1 km Relative Position Actual and Formal Errors/5 Second Update.

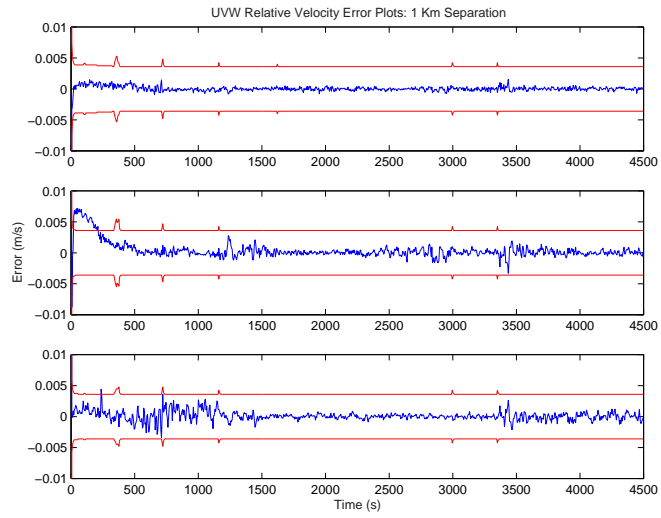


Figure 7.21: Best Case 1 km Relative Velocity Actual and Formal Errors/5 Second Update.

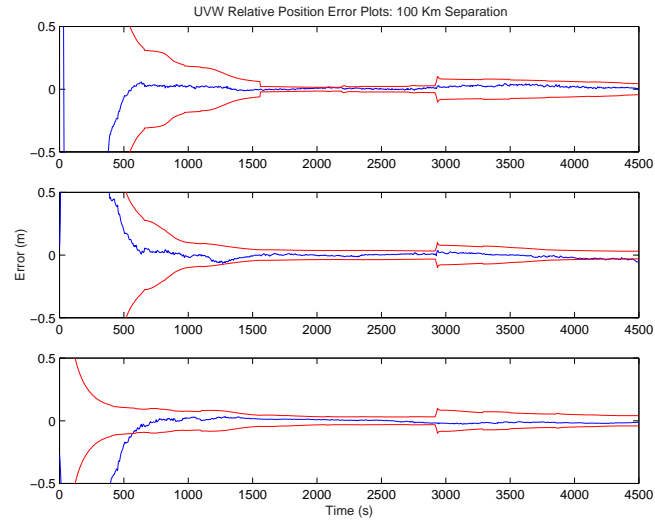


Figure 7.22: Best Case 100 km Relative Position Actual and Formal Errors/5 Second Update.

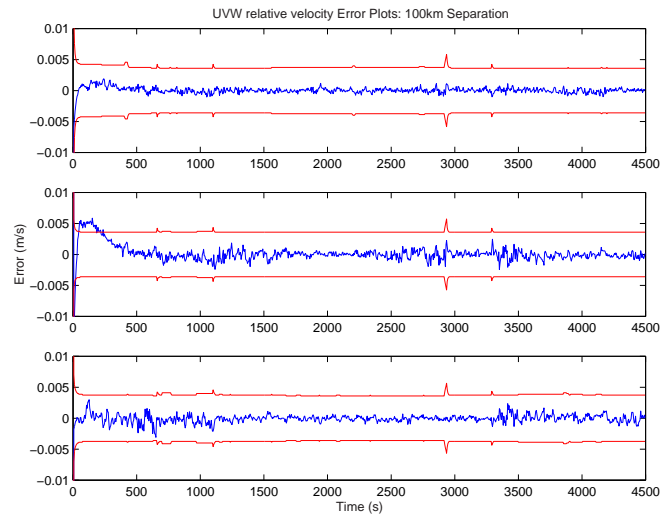


Figure 7.23: Best Case 100 km Relative Velocity Actual and Formal Errors/5 Second Update.

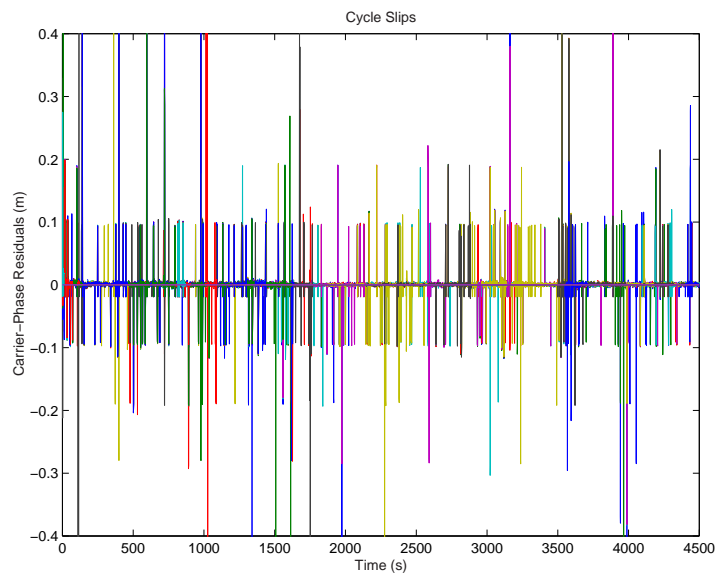


Figure 7.24: Cycle Slip Visualization.

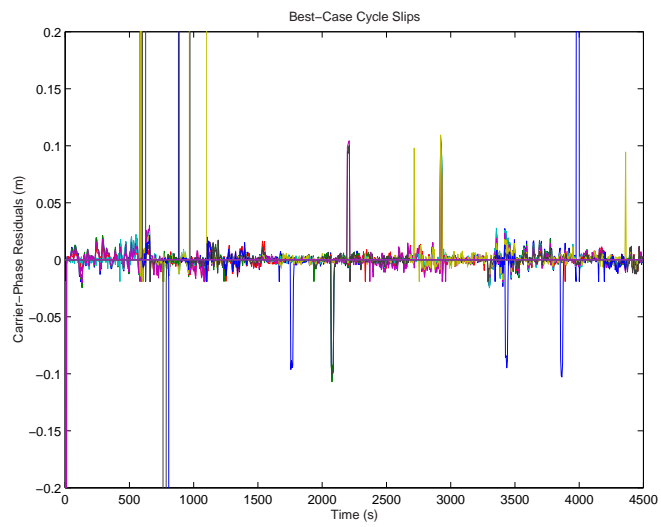


Figure 7.25: Best Case Cycle Slip Profile.

Table 7.9: Relative State Error: Best Case Filter Runs (mean \pm std)

State		1 km Case	100 km Case
1 Second			
Δu	m	-0.0018 ± 0.0034	0.0062 ± 0.0197
Δv		-0.0015 ± 0.0026	-0.0292 ± 0.0157
Δw		-0.0053 ± 0.0025	-0.0030 ± 0.0123
$\Delta \dot{u}$	m/s	0.0000 ± 0.0001	-0.0000 ± 0.0002
$\Delta \dot{v}$		-0.000 ± 0.0002	0.0000 ± 0.0003
$\Delta \dot{w}$		0.0000 ± 0.0002	0.0000 ± 0.0001
5 Second			
Δu	m	-0.0025 ± 0.0029	0.0131 ± 0.0122
Δv		-0.0033 ± 0.0024	-0.0042 ± 0.0155
Δw		-0.0097 ± 0.0008	-0.0069 ± 0.0124
$\Delta \dot{u}$	m/s	0.0000 ± 0.0002	0.0001 ± 0.0003
$\Delta \dot{v}$		-0.0000 ± 0.0005	-0.0006 ± 0.0006
$\Delta \dot{w}$		0.00000 ± 0.0002	0.0003 ± 0.0004

Table 7.9 illustrates the filter performance when the cycle slips are kept to a minimum, as depicted in Figure 7.25. These accuracies range from two to five times lower than the ensemble runs, which combined data with varying amounts of cycle slips. Accordingly, in order to maximize the filter’s efficiency, GPS receivers which have a minimal cycle slip rate should be chosen.

7.1.5 Receiver Clock Model Validation

As described in Chapter 4, the GPS receiver clock and receiver offsets are modelled as random variables that are disconnected. Traditionally, the frequency offset is modelled as the derivative of the clock offset state. The clock steering performed by the Orion disconnects these states, which requires

them to be modelled independently. Figures 7.26 and 7.27 show the typical estimation values for the two clock parameters. It is apparent that there is little correlation between the figures, which validates the modelling approach.

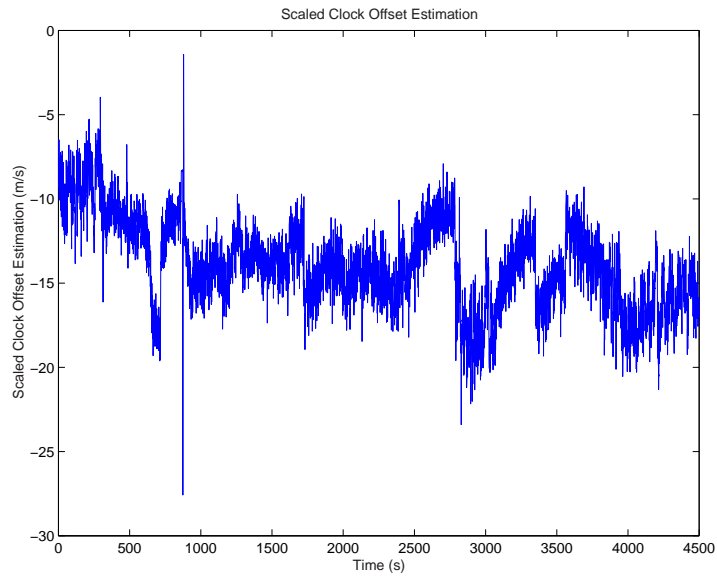


Figure 7.26: GPS Clock Offset Estimation.

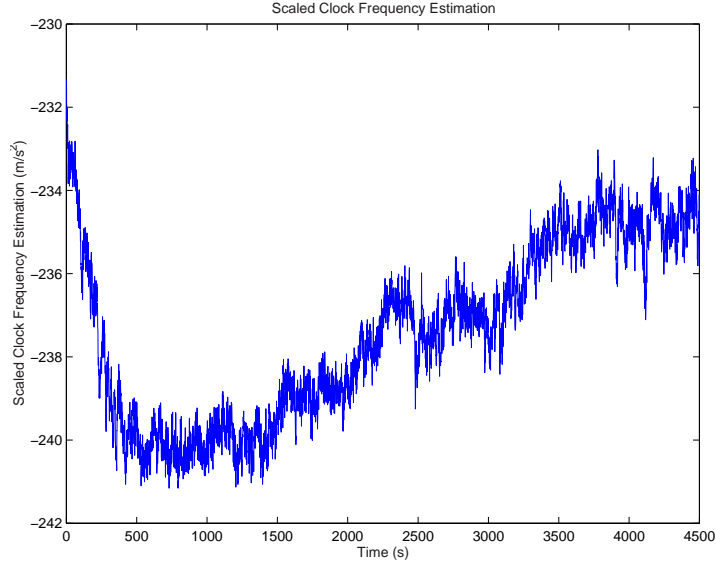


Figure 7.27: GPS Frequency Offset Estimation.

7.2 Extended Formation Flying Testbed Results

In order to test robustness of the navigation and control routines, they were inserted into the EFFTb communication package. The fidelity of the algorithms were tested by requiring them to process real time data at a 5 second update rate, and generate control vectors at 300 second intervals. A benchmark test was performed in the GPS Laboratory at UT/CSR. This result was then used to analyze the results generated from tests spanning the internet.

7.2.1 Two Satellite Local Simulation

Since the GPS simulator at UT/CSR is limited to two spacecraft, initial EFFTb tests were conducted with one hub spacecraft and one node craft. This testing allowed for the verification of the communication links, and the validation of the filter and controller in the real-time environment. Figure 7.28

and Figure 7.29 demonstrate the relative position and velocity errors, which are summarized in Table 7.11.

Table 7.10: Relative State Error: EFFTb 2 Satellite Simulation (mean \pm std)

State		Steady State Error
Δu	(m)	0.0435 ± 0.0621
Δv		0.0300 ± 0.0345
Δw		-0.0126 ± 0.0604
$\Delta \dot{u}$	(m/s)	-0.0000 ± 0.0002
$\Delta \dot{v}$		-0.0001 ± 0.0003
$\Delta \dot{w}$		0.0000 ± 0.0004

The filter performs very closely to the predicted laboratory results. Excluding the large jump at 3500 seconds, the steady state noise is around 3 cm. The spike in estimation error is caused by a large (35 cm/s) maneuver commanded by the controller. The filter treats the delta-v as an impulse, while the truth environment integrates the maneuver over one second. Additionally, the large maneuver causes the receiver to lose lock on all of the GPS satellites, so the filter processes no relative measurements for 10 seconds while it resets the ambiguity terms. During these 10 seconds, the filter's dynamic model and the truth model drift apart, yielding the error shown in the plot. It is also shown that within 400 seconds, the solution has once again converged.

The tracking errors are shown in Figure 7.30. The controller performed as in the offline cases, exhibiting a 20 meter tracking error over the simulation period. Figure 7.31 illustrates the control effort generated by the controller. The maneuvers are typically on the order of 5 cm/s in any direction, but can grow as high as 30 cm/s. It is apparent that such a control scheme would be very costly in fuel, and may not be an optimal control strategy.

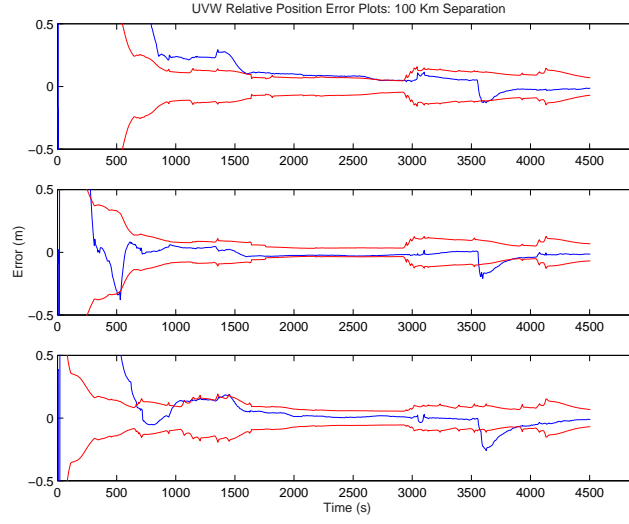


Figure 7.28: Ensemble Mean Relative Position Error Actual and Formal Errors at 100 km.

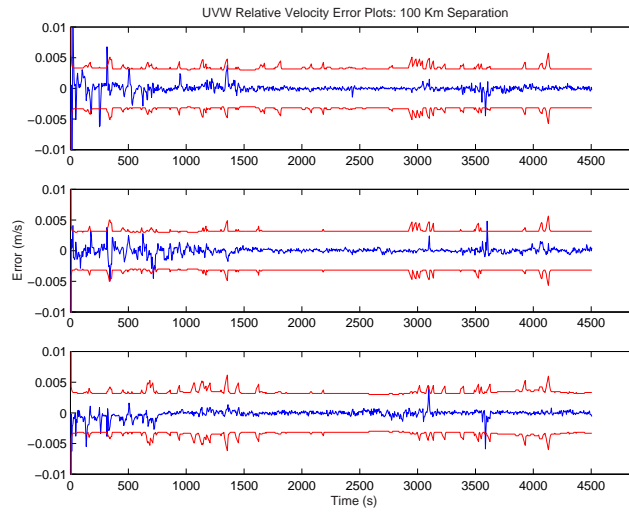


Figure 7.29: Ensemble Mean Relative Velocity Actual and Formal Errors at 100 km.

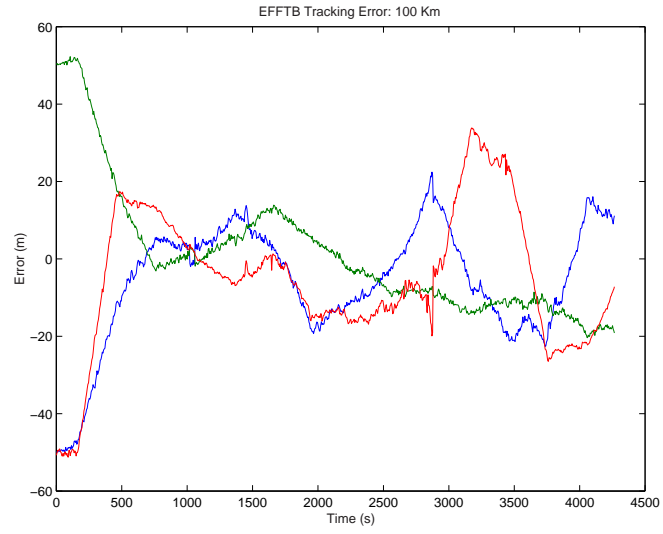


Figure 7.30: Ensemble Mean Tracking Error with 300 Sec Window

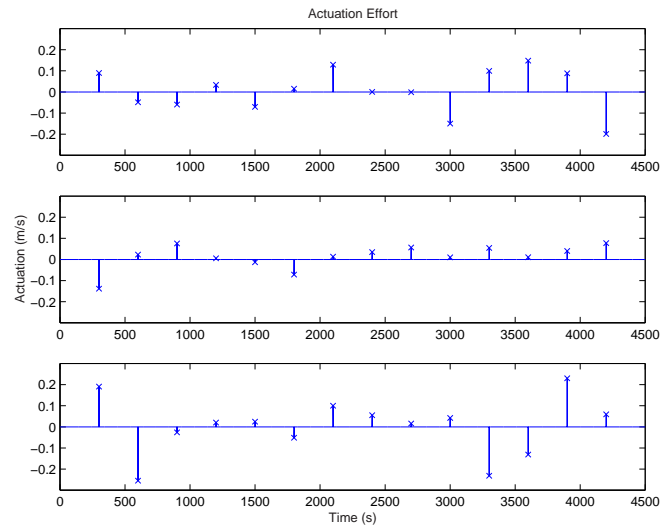


Figure 7.31: Ensemble Mean Actuation Effort in the $\hat{u}, \hat{v}, \hat{w}$ frame

The controller passes the maneuver back to the filter to update its dynamical model. Since no error is superimposed upon this velocity vector, the covariance envelope is not widened during the burns. This can be visualized in Figure 7.28, especially around the 3600 second mark. A degree of uncertainty should be added to the solution to compensate for the the misrepresentation of the maneuver, requiring an increase of the covariance around the control time.

7.2.2 Multi-Satellite Formation

The full EFFTb was tested by coupling the Spirent GPS signal generator located at UT/CSR with the 4 RF signal simulator hosted at Goddard Space Flight Center’s Formation Flying Testbed. Several small modifications were performed on the communication environment in order to comply with network security policies at both sites.

The formation scenario implemented for the final phase of testing was similar to the set of conditions presented earlier. The four node spacecraft were distributed approximately 100 km from the hub spacecraft in either a leading or trailing position. The results of four real-time data sets were averaged to generate Figures 7.32 and 7.33. Both the relative position and velocity errors were consistent with the previous results, implying that it is not necessary to maintain exact synchronization between the two simulation environments. The typical offset between the two environment computers was on the order of one second, well below the 5 second filter update rate. For applications requiring estimates closer to 1Hz, time synchronization devices would be required at both laboratories.

A set of four simulations was run with the controller enabled. The

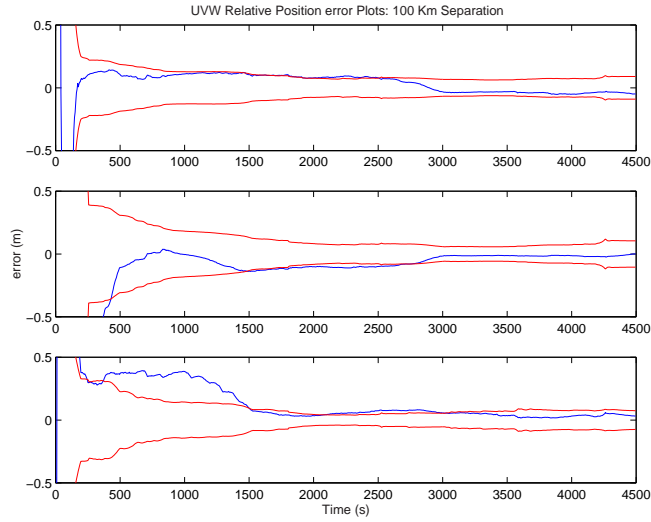


Figure 7.32: Full Simulation Environment Ensemble Mean Relative Position Actual and Formal Errors

scenario contained five satellites, the hub and four node spacecraft. Such a situation would have been previously impossible without a 5 RF GPS simulator. The controller was again set to burn every 300 seconds, and the initial offsets from the desired trajectory were also held the same. The relative position and velocity errors are illustrated in Figures 7.34 and 7.35 respectively. It is apparent that the ensemble averaging of the 16 individual runs helped to smooth out many of the large jumps seen in Figure 7.28. The tracking error shown in 7.36, also demonstrates a somewhat smoother nature. The EFFTb communication system had no difficulties in routing messages to any of the nodes, and seamlessly linked the two laboratories.

Figure 7.37 illustrates the average control effort exhausted over the four controlled simulations. The total control effort is detailed in Table 7.12. On average, it would require three meters/second of control per orbit to maintain

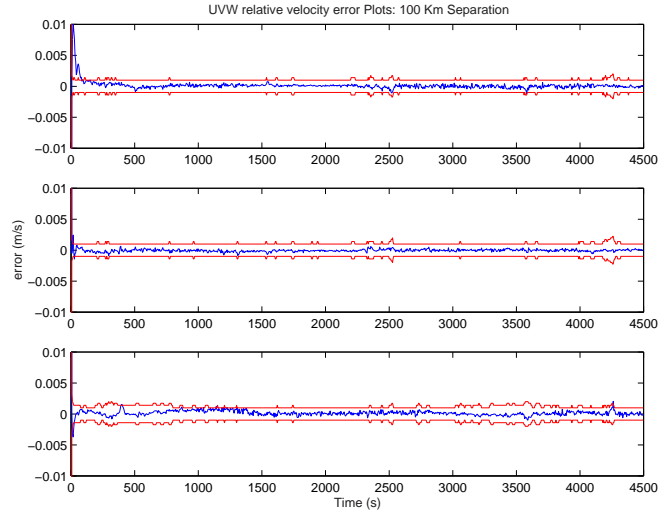


Figure 7.33: Full Simulation Environment Ensemble Mean Relative Velocity Actual and Formal Errors

the desired spacing of this orbit. This expenditure would be prohibitively high for most formation missions. The point of this example was to demonstrate a proof of control capability using this navigation system rather than to recommend a specific guidance method, which would be application dependent.

Table 7.11: Relative State Error: EFFTb 5 Satellite Simulation (mean \pm std)

Non-Controlled State		Steady State Error
Δu	(m)	0.0114 ± 0.0699
Δv		-0.0559 ± 0.0458
Δw		0.0323 ± 0.0312
$\Delta \dot{u}$	(m/s)	0.00000 ± 0.0002
$\Delta \dot{v}$		-0.00002 ± 0.0001
$\Delta \dot{w}$		0.00006 ± 0.0003
Controlled State		Steady State Error
Δu	(m)	-0.0612 ± 0.0710
Δv		-0.0943 ± 0.0556
Δw		0.0037 ± 0.0962
$\Delta \dot{u}$	(m/s)	0.00000 ± 0.0005
$\Delta \dot{v}$		0.00001 ± 0.0004
$\Delta \dot{w}$		0.00016 ± 0.0007

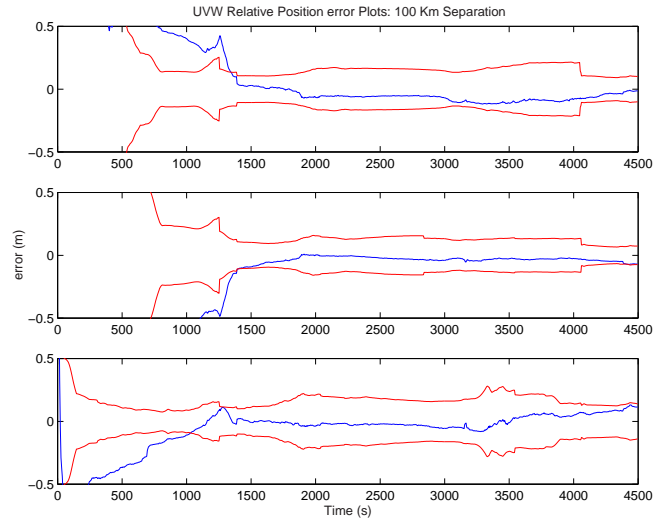


Figure 7.34: Ensemble Mean Relative Position Actual and Formal Errors With Control Effort

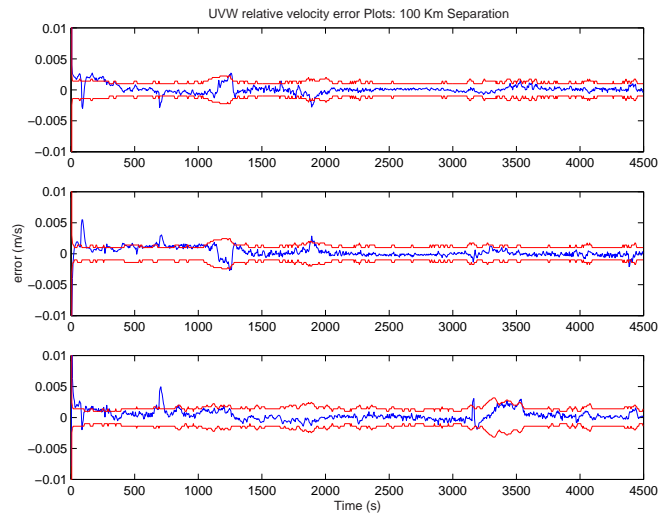


Figure 7.35: Ensemble Mean Relative Velocity Actual and Formal Errors With Control Effort

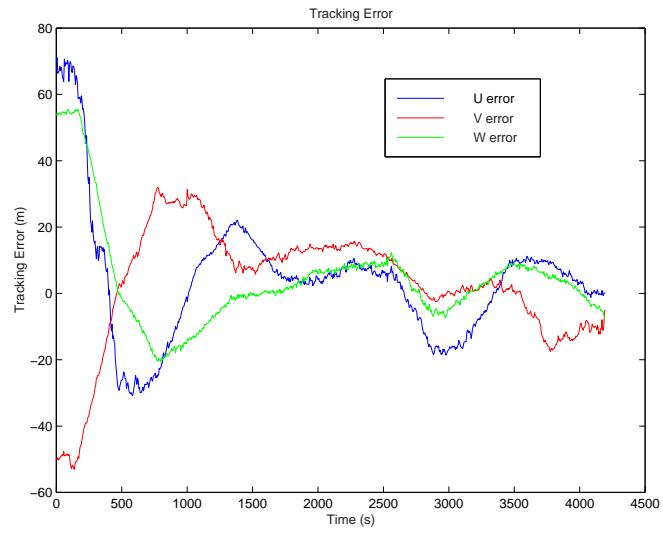


Figure 7.36: Ensemble Mean Tracking Error
in the $\hat{u}, \hat{v}, \hat{w}$ frame

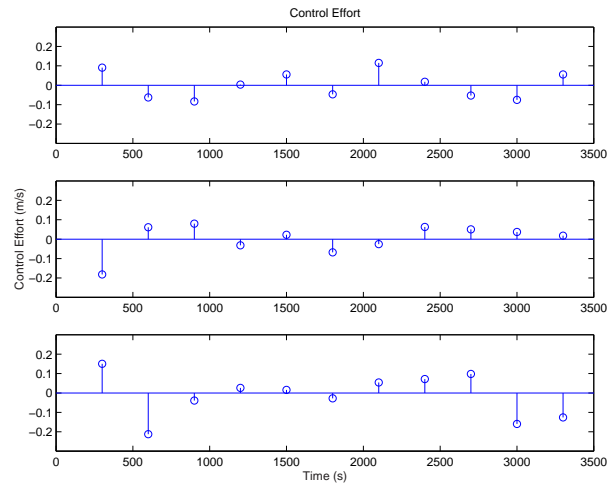


Figure 7.37: Ensemble Mean Actuation Effort
in the $\hat{u}, \hat{v}, \hat{w}$ frame

Table 7.12: Control Effort in the $\hat{u}, \hat{v}, \hat{w}$ frame

State		Control Effort
u	m/s	0.8952
v		0.777885
w		1.144059
<i>Total</i>		2.817145

Chapter 8

Conclusion

The purpose of this work was to develop a general formation navigation filter capability. The filter possesses high accuracy relative position estimation over a variety of inter-satellite distances and number of satellites included in the formation. This filter was tested in a real-time closed-loop hardware-in-the-loop testbed which was developed to utilize the internet to exchange measurements. A summary of the results is presented in the following section. Finally, directions for future work are indicated.

8.1 Summary of Results

The study of large satellite formations can be divided into two categories: formations with large intersatellite spacings and formations with a large number of satellites. It was desired to look at both of these problems as generally as possible, and show that both are feasible for future formation missions using GPS sensor measurements.

The problem of relative navigation over large baselines was examined first. In order to strike a balance between filter complexity and accuracy, a plant model with atmospheric drag and J2 gravity perturbation was implemented. All of the GPS observables were used in a filter state that estimated both the relative position and the absolute position of the host satellite.

It was found that the filter performed quite well over a large range of

baselines. It was determined that for shorter baselines, the accuracy of the absolute position had little impact on the overall relative solution. For the relative solution at the 1Hz update rate, both the 1 km separation and the 100 km separation had steady state noise of less than a few centimeters over an ensemble of runs. The 1 km results were consistent with other works, validating the filter design, and displaying its robustness at multiple baselines. At 500 km, the models, primarily the ionospheric delay model, began to degrade in such a way that the integer ambiguity states became difficult to determine. This led to a drop in performance, signaling the limit of the design outlined here.

When the measurement update rate was increased to 5 seconds, neither the 1 km nor the 100 km estimator accuracies decreased significantly. In general, the steady state errors doubled, but left them still below 10 centimeters.

One of the problems that was faced in processing the actual GPS data was the handling of cycle slips. The slips occurred sporadically in some of the receivers that were used in the testing, and more frequently in others. Simulations with a large number of slips had a much slower relative solution convergence rate, and higher overall navigation error. It was shown that for a run with minimal cycle slips, the 1 km spacing had steady state error no larger than 1 centimeter, and the 100 km case maintained error less than 2 centimeters. These results validate the estimator design and emphasize the importance of the hardware performance on the final results.

A simple linear controller was developed in order to close the loop in the EFFTb. The purpose of the controller was to demonstrate the closed-loop hardware-in-the-loop system performance for a representative formation flying mission. The designed controller was able to accurately control the system with no feedback error, and when the controller update interval was reduced

to 10 seconds, it was able to track small amounts of measurement error precisely. When the absolute state filter feedback was simulated by adding 3 meters of feedback noise, the accuracy of the controller dropped to 20 meters. This was determined to be the result of the less accurate absolute position solution being propagated forward in time to generate the desired trajectory. It was shown that if the absolute position error was reduced to .5 meters, the controller accuracy improved to within a couple of meters.

The Extended Formation Flying Testbed (EFFTB) was developed to link hardware from remote sites in order to simulate formations with many satellites. This was done because it is unrealistic to locate all of the testing resources that are required for a large formation at one facility. The communication package was designed to allow for many combinations of hardware or software components required for testing. Once developed, it was specialized to accommodate Orion GPS receivers and the Spirent GPS signal simulators used in this test program. This instance of the EFFTB makes use of several different programs running simultaneously on different machines to emulate a satellite formation in orbit. Data is passed back and forth in a broadcast fashion to mimic space communications. Every effort was made to keep the simulation environment as authentic as possible.

The long baseline filter and the linear controller were inserted into the testbed architecture in order to verify the offline results. Since the 500 km baseline demonstrated to be at the limit of applicability for this filter, the focus was placed on the 100 km filter. It was found that the steady state error was on the order of 5 centimeters, only 3 times as noisy as the offline case. The decrease in performance is due to the addition of the controller, which caused small jumps in the relative position error when the control effort was large. The tracking error had a 20 meter error band, as predicted in the offline

testing. Both the controller and the estimator performed comparably in the real-time testing and in the offline studies, verifying the design and utility of this navigation system.

There was no appreciable decrease in accuracy when the simulation environment was extended to include other GPS simulators located at a different facility. The synchronization between the two simulation systems did not play a critical role in the relative navigation, assuming the update rate was greater than the anticipated time offset. Using the newly-developed tool, closed-loop simulations can be created which span multiple environments, allowing the study of navigation and control on truly large formations.

8.2 Future Work

The results presented in the previous section suggest that the filter design is robust up to 500 km, but a few simplicities were included that may increase the errors from the values stated here.

8.2.1 Ionosphere Study

Future work should be done examining the effects of the ionosphere, especially at longer baselines. The Spirent simulator can produce various smooth ionosphere models, which may not accurately represent the actual space environment. This filter design should be tested against a more realistic ionosphere model to further examine its effects on the ambiguity resolution.

8.2.2 Multi-Path Study

GPS multi-path error was neglected during this study, but could serve as a major error source. This error is highly dependent on the satellite's shape, and cannot easily be modelled in a general sense. It may be helpful to take a single satellite, and analyze the impact that multi-path would have on it, generating some insight into the effect on the filter's performance.

8.2.3 Attitude Modelling and Control

The main piece that is missing from this formation simulation is the attitude determination and control capabilities. It was assumed that the control was always performed along the X,Y,Z inertial axis, regardless of the satellite's orientation. The inclusion of attitude modelling would open a new realm of reality to the testbed. This work is anticipated to be performed in the near future at UT/CSR.

8.2.4 Formation Control

The EFFTb can be used as a tool to investigate the control of a formation as a single entity. The different control architectures can be studied to determine which is most appropriate for a given mission. With the addition of more accurate thruster models, formation fuel consumption and mission life can also be studied.

8.2.5 EFFTb Extension

The EFFTb was designed to be a general purpose simulation tool. Future work could include the developing of different instances that use other simulated measurements instead of the GPS simulator/receiver combination. In doing so, missions such as High Earth Orbit (HEO) or deep space missions can be considered and modelled. The last step could be to connect flight computers to the EFFTb in order to encompass the limited computation power experienced in space.

Appendices

Appendix A

Dynamic Equation Partial Derivatives

A.1 Dynamic Equation Partial Derivatives

$$\begin{array}{lll}
 \frac{\partial \dot{x}}{\partial x} = 0 & \frac{\partial \dot{x}}{\partial x} = 0 & \frac{\partial \dot{x}}{\partial x} = 0 \\
 \frac{\partial \dot{y}}{\partial x} = 0 & \frac{\partial \dot{y}}{\partial y} = 0 & \frac{\partial \dot{y}}{\partial z} = 0 \\
 \frac{\partial \dot{z}}{\partial x} = 0 & \frac{\partial \dot{z}}{\partial y} = 0 & \frac{\partial \dot{z}}{\partial z} = 0 \\
 \frac{\partial \dot{y}}{\partial x} = 0 & \frac{\partial \dot{z}}{\partial x} = 0 & \frac{\partial \dot{z}}{\partial y} = 0 \\
 \frac{\partial \dot{x}}{\partial y} = 0 & \frac{\partial \dot{x}}{\partial z} = 0 & \frac{\partial \dot{y}}{\partial z} = 0
 \end{array}$$

$$\begin{array}{ll}
 \frac{\partial \dot{x}}{\partial \dot{x}} = 1 \\
 \frac{\partial \dot{y}}{\partial \dot{y}} = 1 \\
 \frac{\partial \dot{z}}{\partial \dot{z}} = 1
 \end{array}$$

$$\begin{aligned}
 \frac{\partial \ddot{x}}{\partial x} = & -\frac{\mu}{r^3} \left[1 - \frac{3}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(5 \frac{z^2}{r^2} - 1 \right) \right] \\
 & + 3 \frac{\mu x^2}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 1 \right) \right] \\
 & + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{x(\dot{x} + \omega_E y)}{r H} \\
 & + \frac{1}{2} C_D \frac{A}{m} \rho \frac{\omega_E (\dot{x} + \omega_E y)(\dot{y} - \omega_E x)}{V}
 \end{aligned}$$

$$\begin{aligned}
\frac{\partial \ddot{x}}{\partial y} &= 3 \frac{\mu xy}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 1 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{y(\dot{x} + \omega_E y)}{rH} \\
&\quad - \frac{1}{2} C_D \frac{A}{m} \rho \left[V \omega_E + \frac{\omega_E (\dot{x} + \omega_E y)^2}{V} \right] \\
\frac{\partial \ddot{x}}{\partial z} &= 3 \frac{\mu xz}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 3 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{z(\dot{x} + \omega_E y)}{rH} \\
\frac{\partial \ddot{x}}{\partial \dot{x}} &= -\frac{1}{2} C_D \frac{A}{m} \rho \left[V + \frac{(\dot{x} + \omega_E y)^2}{V} \right] \\
\frac{\partial \ddot{x}}{\partial \dot{y}} &= -\frac{1}{2} C_D \frac{A}{m} \rho \frac{(\dot{y} - \omega_E x)(\dot{x} + \omega_E y)}{V} \\
\frac{\partial \ddot{x}}{\partial \dot{z}} &= -\frac{1}{2} C_D \frac{A}{M} \rho \frac{\dot{z}(\dot{x} + \omega_E y)}{V} \\
\frac{\partial \ddot{y}}{\partial x} &= 3 \frac{\mu xy}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 1 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{x(\dot{y} - \omega_E x)}{rH} \\
&\quad - \frac{1}{2} C_D \frac{A}{m} \rho \left[V \omega_E - \frac{\omega_E (\dot{y} - \omega_E x)^2}{V} \right] \\
\frac{\partial \ddot{y}}{\partial y} &= -\frac{\mu}{r^3} \left[1 - \frac{3}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(5 \frac{z^2}{r^2} - 1 \right) \right] \\
&\quad + 3 \frac{\mu y^2}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 1 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{y(\dot{y} - \omega_E x)}{rH} \\
&\quad - \frac{1}{2} C_D \frac{A}{m} \rho \frac{\omega_E (\dot{x} + \omega_E y)(\dot{y} - \omega_E x)}{V}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \ddot{y}}{\partial z} &= 3 \frac{\mu y z}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 3 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{z(\dot{y} - \omega_E y)}{r H} \\
\frac{\partial \ddot{y}}{\partial \dot{x}} &= -\frac{1}{2} C_D \frac{A}{m} \rho \frac{(\dot{y} - \omega_E x)(\dot{x} + \omega_E y)}{V} \\
\frac{\partial \ddot{y}}{\partial \dot{y}} &= -\frac{1}{2} C_D \frac{A}{m} \rho \left[V + \frac{(\dot{y} - \omega_E x)^2}{V} \right] \\
\frac{\partial \ddot{y}}{\partial \dot{z}} &= -\frac{1}{2} C_D \frac{A}{M} \rho \frac{\dot{z}(\dot{y} + \omega_E x)}{V} \\
\frac{\partial \ddot{z}}{\partial x} &= 3 \frac{\mu x z}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 3 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{\dot{z} x}{r H} \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho \frac{\omega_E \dot{z}(\dot{y} - \omega_E x)}{V} \\
\frac{\partial \ddot{z}}{\partial y} &= 3 \frac{\mu y z}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 3 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{\dot{z} y}{r H} \\
&\quad - \frac{1}{2} C_D \frac{A}{m} \rho \frac{\omega_E \dot{z}(\dot{x} - \omega_E y)}{V} \\
\frac{\partial \ddot{z}}{\partial z} &= -\frac{\mu}{r^3} \left[1 - \frac{3}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(5 \frac{z^2}{r^2} - 3 \right) \right] \\
&\quad + 3 \frac{\mu z^2}{r^5} \left[1 - \frac{5}{2} J_2 \left(\frac{R_E}{r} \right)^2 \left(7 \frac{z^2}{r^2} - 5 \right) \right] \\
&\quad + \frac{1}{2} C_D \frac{A}{m} \rho V \frac{z \dot{z}}{r H}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \ddot{z}}{\partial \dot{x}} &= -\frac{1}{2}C_D \frac{A}{m} \rho \frac{\dot{z}(\dot{x} + \omega_E y)}{V} \\
\frac{\partial \ddot{z}}{\partial \dot{y}} &= -\frac{1}{2}C_D \frac{A}{m} \rho \frac{\dot{z}(\dot{y} + \omega_E x)}{V} \\
\frac{\partial \ddot{z}}{\partial \dot{z}} &= -\frac{1}{2}C_D \frac{A}{m} \rho \left[V + \frac{\dot{z}^2}{V} \right]
\end{aligned}$$

Appendix B

Measurement Equation Partial Derivatives

B.1 Pseudorange Partial Derivatives

$$\begin{array}{lll}
 \frac{\partial P}{\partial \dot{x}} = 0 & \frac{\partial P}{\partial \dot{y}} = 0 & \frac{\partial P}{\partial \dot{z}} = 0 \\
 \frac{\partial P}{\partial \Delta x} = 0 & \frac{\partial P}{\partial \Delta y} = 0 & \frac{\partial P}{\partial \Delta z} = 0 \\
 \frac{\partial P}{\partial \Delta \dot{x}} = 0 & \frac{\partial P}{\partial \Delta \dot{y}} = 0 & \frac{\partial P}{\partial \Delta \dot{z}} = 0 \\
 \frac{\partial P}{\partial b} = 0 & \frac{\partial P}{\partial N} = 0 & \frac{\partial P}{\partial B} = 0
 \end{array}$$

$$\begin{aligned}
 \frac{\partial P}{\partial x} &= -\frac{(r_{x,GPS} - r_x)}{|r_{GPS} - \underline{r}|} \\
 \frac{\partial P}{\partial y} &= -\frac{(r_{y,GPS} - r_y)}{|r_{GPS} - \underline{r}|} \\
 \frac{\partial P}{\partial z} &= -\frac{(r_{z,GPS} - r_z)}{|r_{GPS} - \underline{r}|} \\
 \frac{\partial P}{\partial b} &= 1
 \end{aligned}$$

B.2 Doppler Partial Derivatives

$$\begin{aligned}\frac{\partial D}{\partial \Delta x} &= 0 \\ \frac{\partial D}{\partial \Delta \dot{x}} &= 0 \\ \frac{\partial P}{\partial b} &= 0\end{aligned}$$

$$\begin{aligned}\frac{\partial D}{\partial \Delta y} &= 0 \\ \frac{\partial D}{\partial \Delta \dot{y}} &= 0 \\ \frac{\partial P}{\partial N} &= 0\end{aligned}$$

$$\begin{aligned}\frac{\partial D}{\partial \Delta z} &= 0 \\ \frac{\partial D}{\partial \Delta \dot{z}} &= 0 \\ \frac{\partial P}{\partial B} &= 0\end{aligned}$$

$$\begin{aligned}\frac{\partial D}{\partial x} &= \frac{(\dot{r}_x - \dot{r}_{x,GPS})}{|\underline{r}_{GPS} - \underline{r}|} \\ \frac{\partial D}{\partial y} &= \frac{(\dot{r}_y - \dot{r}_{y,GPS})}{|\underline{r}_{GPS} - \underline{r}|} \\ \frac{\partial D}{\partial z} &= \frac{(\dot{r}_z - \dot{r}_{z,GPS})}{|\underline{r}_{GPS} - \underline{r}|} \\ \frac{\partial D}{\partial \dot{x}} &= \frac{(r_x - r_{x,GPS})}{|\underline{r} - \underline{r}_{GPS}|} \\ \frac{\partial D}{\partial \dot{y}} &= \frac{(r_y - r_{y,GPS})}{|\underline{r} - \underline{r}_{GPS}|} \\ \frac{\partial D}{\partial \dot{z}} &= \frac{(r_z - r_{z,GPS})}{|\underline{r} - \underline{r}_{GPS}|} \\ \frac{\partial D}{\partial \dot{b}} &= 1\end{aligned}$$

B.3 Double-Differenced Carrier Phase Partial Derivatives

$$\begin{array}{lll}
\frac{\partial \lambda \Phi}{\partial \dot{x}} = 0 & \frac{\partial \lambda \Phi}{\partial \dot{y}} = 0 & \frac{\partial \lambda \Phi}{\partial \dot{z}} = 0 \\
\frac{\partial \lambda \Phi}{\partial \Delta \dot{x}} = 0 & \frac{\partial \lambda \Phi}{\partial \Delta \dot{y}} = 0 & \frac{\partial \lambda \Phi}{\partial \Delta \dot{z}} = 0 \\
\frac{\partial \lambda \Phi}{\partial b} = 0 & \frac{\partial \lambda \Phi}{\partial b} = 0 & \frac{\partial \lambda \Phi}{\partial B} = 0
\end{array}$$

$$\begin{aligned}
\frac{\partial \lambda \Phi}{\partial x} &= \left[\frac{-r_{GPS,x}^A + \Delta \underline{r}_x + r_x}{|r_{GPS}^A - \Delta \underline{r} - \underline{r}|} + \frac{(r_{GPS,x}^A - r_x)}{|r_{GPS}^A - \underline{r}|} \right] \\
&\quad - \left[\frac{-r_{GPS,x}^B + r_{\Delta x} + r_x}{|r_{GPS}^B - \Delta \underline{r} - \underline{r}|} + \frac{(r_{GPS,x}^B - r_x)}{|r_{GPS}^B - \underline{r}|} \right] \\
\frac{\partial \lambda \Phi}{\partial y} &= \left[\frac{-r_{GPS,y}^A + \Delta \underline{r}_y + r_y}{|r_{GPS}^A - \Delta \underline{r} - \underline{r}|} + \frac{(r_{GPS,y}^A - r_y)}{|r_{GPS}^A - \underline{r}|} \right] \\
&\quad - \left[\frac{-r_{GPS,y}^B + r_{\Delta y} + r_y}{|r_{GPS}^B - \Delta \underline{r} - \underline{r}|} + \frac{(r_{GPS,y}^B - r_y)}{|r_{GPS}^B - \underline{r}|} \right] \\
\frac{\partial \lambda \Phi}{\partial z} &= \left[\frac{-r_{GPS,z}^A + \Delta \underline{r}_z + r_z}{|r_{GPS}^A - \Delta \underline{r} - \underline{r}|} + \frac{(r_{GPS,z}^A - r_z)}{|r_{GPS}^A - \underline{r}|} \right] \\
&\quad - \left[\frac{-r_{GPS,z}^B + r_{\Delta z} + r_z}{|r_{GPS}^B - \Delta \underline{r} - \underline{r}|} + \frac{(r_{GPS,z}^B - r_z)}{|r_{GPS}^B - \underline{r}|} \right] \\
\frac{\partial \lambda \Phi}{\partial \Delta x} &= \left[\frac{-r_{GPS,x}^A + \Delta r_x + r_x}{|-r_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] \\
&\quad - \left[\frac{-r_{GPS,x}^B + \Delta r_x + r_x}{|-r_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right]
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \lambda \Phi}{\partial \Delta y} &= \left[\frac{-r_{GPS,y}^A + \Delta r_y + r_y}{|-r_{GPS} + \Delta \underline{r} + \underline{r}|} \right] \\
&\quad - \left[\frac{-r_{GPS,y}^B + \Delta r_y + r_y}{|-r_{GPS}^B + \Delta \underline{r}_y + \underline{r}_y|} \right] \\
\frac{\partial \lambda \Phi}{\partial \Delta z} &= \left[\frac{-r_{GPS,z}^A + \Delta r_z + r_z}{|-r_{GPS} + \Delta \underline{r} + \underline{r}|} \right] \\
&\quad - \left[\frac{-r_{GPS,z}^B + \Delta r_z + r_z}{|-r_{GPS}^B + \Delta \underline{r}_z + \underline{r}_z|} \right] \\
\frac{\partial \lambda \Phi}{\partial \lambda N} &= 1
\end{aligned}$$

B.4 Double-Differenced Doppler Partial Derivatives

$$\frac{\partial D_{IJ}^{AB}}{\partial b} = 0 \quad \frac{\partial D_{IJ}^{AB}}{\partial \dot{b}} = 0 \quad \frac{\partial D_{IJ}^{AB}}{\partial N} = 0 \quad \frac{\partial D_{IJ}^{AB}}{\partial B} = 0$$

$$\begin{aligned} \frac{\partial D_{IJ}^{AB}}{\partial x} &= \frac{-\dot{r}_{GPS,x}^A + \Delta \dot{r}_x + \dot{r}_x}{(|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)} \\ &+ \frac{-r_{GPS,x}^A + \Delta r_x + r_x}{(|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)^3} (-\dot{\underline{r}}_{GPS}^A + \Delta \dot{\underline{r}} + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}) \\ &- \left[\frac{-r_{GPS,x}^A + r_x}{-|\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}|} + \frac{r_{GPS,x}^A - r_x}{-|\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}|} \right] (-\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^A + \underline{r}) \\ &\frac{-\dot{r}_{GPS,x}^B + \Delta \dot{r}_x + \dot{r}_x}{(|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)} \\ &+ \frac{-r_{GPS,x}^B + \Delta r_x + r_x}{(|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)^3} (-\dot{\underline{r}}_{GPS}^B + \Delta \dot{\underline{r}} + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}) \\ &- \left[\frac{-r_{GPS,x}^B + r_x}{-|\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}|} + \frac{r_{GPS,x}^B - r_x}{-|\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}|} \right] (-\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^B + \underline{r}) \\ \frac{\partial D_{IJ}^{AB}}{\partial y} &= \frac{-\dot{r}_{GPS,y}^A + \Delta \dot{r}_y + \dot{r}_y}{(|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)} \\ &+ \frac{-r_{GPS,y}^A + \Delta r_y + r_y}{(|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)^3} (-\dot{\underline{r}}_{GPS}^A + \Delta \dot{\underline{r}} + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}) \\ &- \left[\frac{-r_{GPS,y}^A + r_y}{-|\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}|} + \frac{r_{GPS,y}^A - r_y}{-|\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}|} \right] (-\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^A + \underline{r}) \\ &\frac{-\dot{r}_{GPS,y}^B + \Delta \dot{r}_y + \dot{r}_y}{(|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)} \\ &+ \frac{-r_{GPS,y}^B + \Delta r_y + r_y}{(|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)^3} (-\dot{\underline{r}}_{GPS}^B + \Delta \dot{\underline{r}} + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}) \\ &- \left[\frac{-r_{GPS,y}^B + r_y}{-|\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}|} + \frac{r_{GPS,y}^B - r_y}{-|\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}|} \right] (-\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^B + \underline{r}) \end{aligned}$$

$$\begin{aligned}
\frac{\partial D_{IJ}^{AB}}{\partial z} &= \frac{-\dot{r}_{GPS,z}^A + \Delta \dot{r}_z + \dot{r}_z}{(|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)} \\
&+ \frac{-r_{GPS,z}^A + \Delta r_z + r_z}{(|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)^3} (-\dot{\underline{r}}_{GPS}^A + \Delta \dot{\underline{r}} + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}) \\
&- \left[\frac{-r_{GPS,z}^A + r_z}{-|\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}|} + \frac{r_{GPS,z}^A - r_z}{-|\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}|} \right] (-\dot{\underline{r}}_{GPS}^A + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^A + \underline{r}) \\
&\frac{-\dot{r}_{GPS,z}^B + \Delta \dot{r}_z + \dot{r}_z}{(|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)} \\
&+ \frac{-r_{GPS,z}^B + \Delta r_z + r_z}{(|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)^3} (-\dot{\underline{r}}_{GPS}^B + \Delta \dot{\underline{r}} + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}) \\
&- \left[\frac{-r_{GPS,z}^B + r_z}{-|\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}|} + \frac{r_{GPS,z}^B - r_z}{-|\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}|} \right] (-\dot{\underline{r}}_{GPS}^B + \dot{\underline{r}}) \cdot (-\underline{r}_{GPS}^B + \underline{r}) \\
\frac{\partial D_{IJ}^{AB}}{\partial \dot{x}} &= \left[\frac{-r_{GPS,x}^A + \Delta r_x + r_x}{|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|} - \frac{-r_{GPS,x}^A + r_x}{|\underline{r}_{GPS}^A + \underline{r}|} \right] \\
&- \left[\frac{-r_{GPS,x}^B + \Delta r_x + r_x}{|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|} - \frac{-r_{GPS,x}^B + r_x}{|\underline{r}_{GPS}^B + \underline{r}|} \right] \\
\frac{\partial D_{IJ}^{AB}}{\partial \dot{y}} &= \left[\frac{-r_{GPS,y}^A + \Delta r_y + r_y}{|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|} - \frac{-r_{GPS,y}^A + r_y}{|\underline{r}_{GPS}^A + \underline{r}|} \right] \\
&- \left[\frac{-r_{GPS,y}^B + \Delta r_y + r_y}{|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|} - \frac{-r_{GPS,y}^B + r_y}{|\underline{r}_{GPS}^B + \underline{r}|} \right] \\
\frac{\partial D_{IJ}^{AB}}{\partial \dot{z}} &= \left[\frac{-r_{GPS,z}^A + \Delta r_z + r_z}{|-\underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|} - \frac{-r_{GPS,z}^A + r_z}{|\underline{r}_{GPS}^A + \underline{r}|} \right] \\
&- \left[\frac{-r_{GPS,z}^B + \Delta r_z + r_z}{|-\underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|} - \frac{-r_{GPS,z}^B + r_z}{|\underline{r}_{GPS}^B + \underline{r}|} \right]
\end{aligned}$$

$$\begin{aligned}
\frac{\partial D_{IJ}^{AB}}{\partial \Delta x} &= \frac{-\dot{r}_{GPS,x}^A + \Delta \dot{r}_x + \dot{r}}{(| - \underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)^3} \\
&+ \left[\frac{-r_{GPS,x}^A - \Delta r_x - r_x}{| - \underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] (\dot{r}_{GPS}^A - \Delta \dot{r} - \dot{r}) \cdot (\underline{r}_{GPS}^A - \Delta \underline{r} - \underline{r}) \\
&\frac{-\dot{r}_{GPS,x}^B + \Delta \dot{r}_x + \dot{r}}{(| - \underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)^3} \\
&+ \left[\frac{-r_{GPS,x}^B - \Delta r_x - r_x}{| - \underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right] (\dot{r}_{GPS}^B - \Delta \dot{r} - \dot{r}) \cdot (\underline{r}_{GPS}^B - \Delta \underline{r} - \underline{r}) \\
\frac{\partial D_{IJ}^{AB}}{\partial \Delta y} &= \frac{-\dot{r}_{GPS,y}^A + \Delta \dot{r}_y + \dot{r}}{(| - \underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)^3} \\
&+ \left[\frac{-r_{GPS,y}^A - \Delta r_y - r_y}{| - \underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] (\dot{r}_{GPS}^A - \Delta \dot{r} - \dot{r}) \cdot (\underline{r}_{GPS}^A - \Delta \underline{r} - \underline{r}) \\
&\frac{-\dot{r}_{GPS,y}^B + \Delta \dot{r}_y + \dot{r}}{(| - \underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)^3} \\
&+ \left[\frac{-r_{GPS,y}^B - \Delta r_y - r_y}{| - \underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right] (\dot{r}_{GPS}^B - \Delta \dot{r} - \dot{r}) \cdot (\underline{r}_{GPS}^B - \Delta \underline{r} - \underline{r}) \\
\frac{\partial D_{IJ}^{AB}}{\partial \Delta z} &= \frac{-\dot{r}_{GPS,z}^A + \Delta \dot{r}_z + \dot{r}}{(| - \underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|)^3} \\
&+ \left[\frac{-r_{GPS,z}^A - \Delta r_z - r_z}{| - \underline{r}_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] (\dot{r}_{GPS}^A - \Delta \dot{r} - \dot{r}) \cdot (\underline{r}_{GPS}^A - \Delta \underline{r} - \underline{r}) \\
&\frac{-\dot{r}_{GPS,z}^B + \Delta \dot{r}_z + \dot{r}}{(| - \underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|)^3} \\
&+ \left[\frac{-r_{GPS,z}^B - \Delta r_z - r_z}{| - \underline{r}_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right] (\dot{r}_{GPS}^B - \Delta \dot{r} - \dot{r}) \cdot (\underline{r}_{GPS}^B - \Delta \underline{r} - \underline{r})
\end{aligned}$$

$$\begin{aligned}
\frac{\partial D_{IJ}^{AB}}{\partial \Delta \dot{x}} &= \left[\frac{-r_{GPS,x}^A + \Delta r_x + r_x}{-|r_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] - \left[\frac{-r_{GPS,x}^B + \Delta r_x + r_x}{-|r_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right] \\
\frac{\partial D_{IJ}^{AB}}{\partial \Delta \dot{y}} &= \left[\frac{-r_{GPS,y}^A + \Delta r_y + r_y}{-|r_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] - \left[\frac{-r_{GPS,y}^B + \Delta r_y + r_y}{-|r_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right] \\
\frac{\partial D_{IJ}^{AB}}{\partial \Delta \dot{z}} &= \left[\frac{-r_{GPS,z}^A + \Delta r_z + r_z}{-|r_{GPS}^A + \Delta \underline{r} + \underline{r}|} \right] - \left[\frac{-r_{GPS,z}^B + \Delta r_z + r_z}{-|r_{GPS}^B + \Delta \underline{r} + \underline{r}|} \right]
\end{aligned}$$

Appendix C

Filter Verification Procedures

To ensure proper functionality of the EKF, several tests were performed. These tests and their results are provided for completeness.

C.1 Verification of the Dynamic Partial Derivative Matrix

In order to verify the proper construction of the dynamic partial derivative matrix, its entries are compared to the complimentary quantities generated from taking the finite differences of the non-linear differential equations. From the discussion in Chapter 5, the dynamic partial derivative matrix is generated from:

$$\mathbf{A}(t) = \frac{\partial \mathbf{F}}{\partial \underline{x}}|_{x_0}$$

This can be compared to the approximate value computed from the forward difference formula:

$$\frac{\partial \mathbf{F}}{\partial \underline{x}_i} \approx \frac{f_i(x_1, x_2, \dots, x_i + h, \dots, x_n) - f_i(x_1, x_2, \dots, x_i, \dots, x_n)}{h}$$

where f_i is the i^{th} state element that is being examined, and h is a small number. The non-trivial results of this comparison, performed for a 100 Km satellite separation, are detailed in Table C.1 and Table C.2.

Table C.1: Dynamic Partial Differential Equation Validation

Derivative	A Matrix Entry	Numerical Derivative	Difference
$\frac{\partial \ddot{x}}{\partial x}$	4.43337762E-07	4.43339843E-07	-2.08117E-12
$\frac{\partial \ddot{x}}{\partial y}$	-1.63758144E-06	-1.63757952E-06	-1.92069E-12
$\frac{\partial \ddot{x}}{\partial z}$	-9.26468001E-07	-9.26466901E-07	-1.10075E-12
$\frac{\partial \ddot{x}}{\partial \dot{x}}$	-3.18904346E-10	-3.18917300E-10	1.29539E-14
$\frac{\partial \ddot{x}}{\partial \dot{y}}$	1.96743068E-11	1.96751108E-11	-8.04001E-16
$\frac{\partial \ddot{x}}{\partial \dot{z}}$	-7.35739287E-11	-7.35734433E-11	-4.85365E-16
$\frac{\partial \ddot{y}}{\partial x}$	-1.63757938E-06	-1.63757938E-06	1.30301E-17
$\frac{\partial \ddot{y}}{\partial y}$	3.15532103E-07	3.15532141E-07	-3.82064E-14
$\frac{\partial \ddot{y}}{\partial z}$	8.91018701E-07	8.9101873243E-07	-3.15294E-14
$\frac{\partial \ddot{y}}{\partial \dot{x}}$	1.96743068E-11	1.96751123E-11	-8.05459E-16
$\frac{\partial \ddot{y}}{\partial \dot{y}}$	-3.16468679E-10	-3.1647196713E-10	3.28765E-15
$\frac{\partial \ddot{y}}{\partial \dot{z}}$	6.91435351E-11	6.91432569544E-11	2.78177E-16
$\frac{\partial \ddot{z}}{\partial x}$	-9.26494804E-07	9.26494814E-07	9.75885E-15
$\frac{\partial \ddot{z}}{\partial y}$	8.91045651E-07	8.91045641E-07	1.02093E-14
$\frac{\partial \ddot{z}}{\partial z}$	-7.58869552E-07	-7.58869605E-07	5.29765E-14
$\frac{\partial \ddot{z}}{\partial \dot{x}}$	-7.35739287E-11	-7.35734404E-11	-4.88296E-16
$\frac{\partial \ddot{z}}{\partial \dot{y}}$	6.91435351E-11	6.91432596E-11	2.75518E-16
$\frac{\partial \ddot{z}}{\partial \dot{z}}$	-5.56539317E-10	-5.56536974E-10	-2.34303E-15

Table C.2: Dynamic Partial Differential Equation Validation Cont.

Derivative	A Matrix Entry	Numerical Derivative	Difference
$\frac{\partial \Delta \ddot{x}}{\partial x}$	4.25051620E-07	4.25051613E-07	6.58033E-15
$\frac{\partial \Delta \ddot{x}}{\partial y}$	-1.61680315E-06	-1.61680318E-06	3.25156E-14
$\frac{\partial \Delta \ddot{x}}{\partial z}$	-9.58182648E-07	-9.58182615E-07	-3.27764E-14
$\frac{\partial \Delta \ddot{x}}{\partial \dot{x}}$	-3.20395142E-10	-3.20378886E-10	-1.62555E-14
$\frac{\partial \Delta \ddot{x}}{\partial \dot{y}}$	2.12700884E-11	2.12727721E-11	-2.68365E-15
$\frac{\partial \Delta \ddot{x}}{\partial \dot{z}}$	-7.59046096E-11	-7.58928249E-11	-1.17847E-14
$\frac{\partial \Delta \ddot{y}}{\partial x}$	-1.61680111E-06	-1.61680339E-06	2.28546E-12
$\frac{\partial \Delta \ddot{y}}{\partial y}$	2.92706778E-07	2.92708944E-07	-2.16553E-12
$\frac{\partial \Delta \ddot{y}}{\partial z}$	9.19767672E-07	9.19769109E-07	-1.43745E-12
$\frac{\partial \Delta \ddot{y}}{\partial \dot{x}}$	2.12700884E-11	2.12727633E-11	-2.67489E-15
$\frac{\partial \Delta \ddot{y}}{\partial \dot{y}}$	-3.17854490E-10	-3.17858283E-10	3.79343E-15
$\frac{\partial \Delta \ddot{y}}{\partial \dot{z}}$	7.15372075E-11	7.15292774E-11	7.93010E-15
$\frac{\partial \Delta \ddot{z}}{\partial x}$	-9.58210700E-07	-9.58210693E-07	-6.92870E-15
$\frac{\partial \Delta \ddot{z}}{\partial y}$	9.19795839E-07	9.19795864E-07	-2.43985E-14
$\frac{\partial \Delta \ddot{z}}{\partial \dot{x}}$	-7.17760550E-07	-7.17760552E-07	2.18525E-15
$\frac{\partial \Delta \ddot{z}}{\partial \dot{y}}$	-7.58941111E-11	-7.58930910E-11	-1.02002E-15
$\frac{\partial \Delta \ddot{z}}{\partial \dot{z}}$	7.15267089E-11	7.15293743E-11	-2.66539E-15
$\frac{\partial \Delta \ddot{z}}{\partial x}$	-5.53001301E-10	-5.52997605E-10	-3.69589E-15

C.2 Verification of the Measurement Partial Derivative Matrix

The measurement partial derivative matrix was verified as in Section C.1. The non-trivial results are listed in Table C.3.

Table C.3: Measurement Partial Differential Equation Validation

Derivative	H Matrix Entry	Numerical Derivative	Difference
$\frac{\partial P}{\partial x}$	7.84860376E-01	7.84860260E-01	1.15618E-07
$\frac{\partial P}{\partial y}$	-3.44975293E-01	-3.44975283E-01	-1.00706E-08
$\frac{\partial P}{\partial z}$	-5.14768327E-01	-5.14768323E-01	-4.35421E-09
$\frac{\partial D}{\partial x}$	1.64229470E-04	1.64229470E-04	1.65339E-13
$\frac{\partial D}{\partial y}$	-1.73000219E-04	-1.73000200E-04	-1.96636E-11
$\frac{\partial D}{\partial z}$	3.66335628E-04	3.66335629E-04	-1.16779E-12
$\frac{\partial D}{\partial \dot{x}}$	7.84860260E-01	7.84860260E-01	-5.08993E-12
$\frac{\partial D}{\partial \dot{y}}$	-3.44975283E-01	-3.44975283E-01	-1.92002E-12
$\frac{\partial D}{\partial \dot{z}}$	-5.14768323E-01	-5.14768323E-01	1.13600E-11
$\frac{\partial \delta \lambda \Phi}{\partial x}$	-2.02226288E-03	-2.02246146E-03	1.98578E-07
$\frac{\partial \delta \lambda \Phi}{\partial y}$	6.38032661E-04	6.37983933E-04	4.87274E-08
$\frac{\partial \delta \lambda \Phi}{\partial z}$	5.45557807E-04	5.45673565E-04	-1.15758E-07
$\frac{\partial \delta \lambda \Phi}{\partial \Delta x}$	-1.73277916E+00	-1.73277916E+00	-4.49096E-09
$\frac{\partial \delta \lambda \Phi}{\partial \Delta y}$	1.76946906E-01	1.76946894E-01	1.24717E-08
$\frac{\partial \delta \lambda \Phi}{\partial \Delta z}$	2.38128883E-01	2.38128962E-01	-7.85587E-08
$\frac{\partial \delta D}{\partial x}$	6.16708027E-06	6.16705936E-06	2.09169E-11
$\frac{\partial \delta D}{\partial y}$	-3.09217499E-07	-3.09234916E-07	1.74173E-11
$\frac{\partial \delta D}{\partial z}$	-1.81840626E-06	-1.81841582E-06	9.56467E-12
$\frac{\partial \delta D}{\partial \dot{x}}$	-2.02246147E-03	-2.02246146E-03	-7.15000E-12
$\frac{\partial \delta D}{\partial \dot{y}}$	6.37983912E-04	6.37983933E-04	-2.14815E-11
$\frac{\partial \delta D}{\partial \dot{z}}$	5.45673581E-04	5.45673565E-04	1.56880E-11
$\frac{\partial \delta D}{\partial \Delta x}$	-2.44516126E-04	-2.44516132E-04	5.69595E-12
$\frac{\partial \delta D}{\partial \Delta y}$	4.78947935E-06	4.78946488E-06	1.44737E-11
$\frac{\partial \delta D}{\partial \Delta z}$	2.90376513E-05	2.90376732E-05	-2.18656E-11
$\frac{\partial \delta D}{\partial \Delta \dot{x}}$	-1.73277918E+00	-1.73277918E+00	6.41998E-12
$\frac{\partial \delta D}{\partial \Delta \dot{y}}$	1.76946888E-01	1.76946888E-01	-5.44001E-12
$\frac{\partial \delta D}{\partial \Delta \dot{z}}$	2.38128946E-01	2.38128946E-01	-5.42000E-12

C.3 Verification of Measurement Noise Settings

Verification of the measurement noise settings in the filter was achieved by comparing the standard deviation on the measurement residuals to the one-sigma setting used in the filter. Figure C.1 illustrates the residuals over one orbit, and Table C.4 provides the comparison of the filter settings.

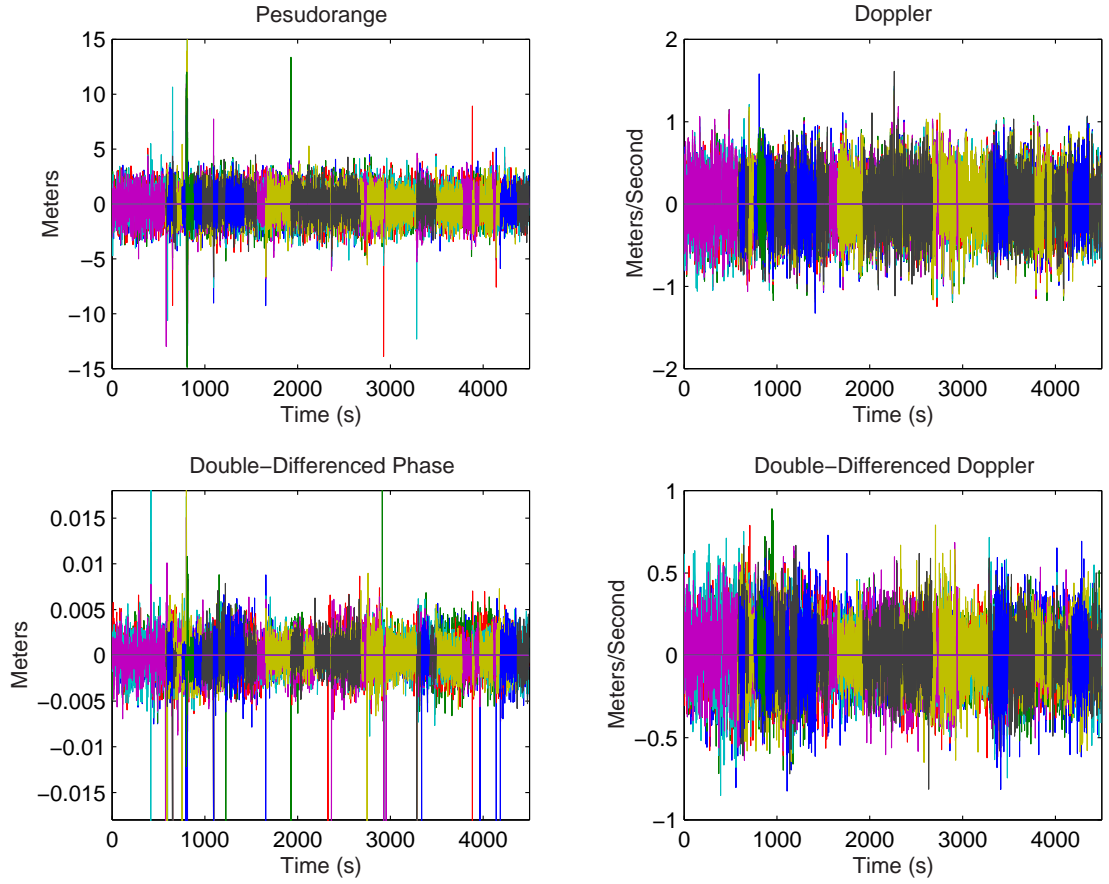


Figure C.1: Measurement Residuals

Table C.4: Measurement Noise Verification.		
Measurement	Residuals	Filter Setting
P(m)	0.8099	1
D(m/s)	0.1981	.25
$\delta\lambda\Phi$ (m)	0.0012	.004
δD (m/s)	0.1055	.15

C.4 Covariance Update Verification

In order to verify the covariance update process was reducing the covariance, it was plotted before and after the update. A typical result for the absolute case are depicted in Figure C.2. Figure C.3 illustrates a relative position covariance update. In both sets of figures, the blue dots are pre-update covariances and the red dots are after the update.

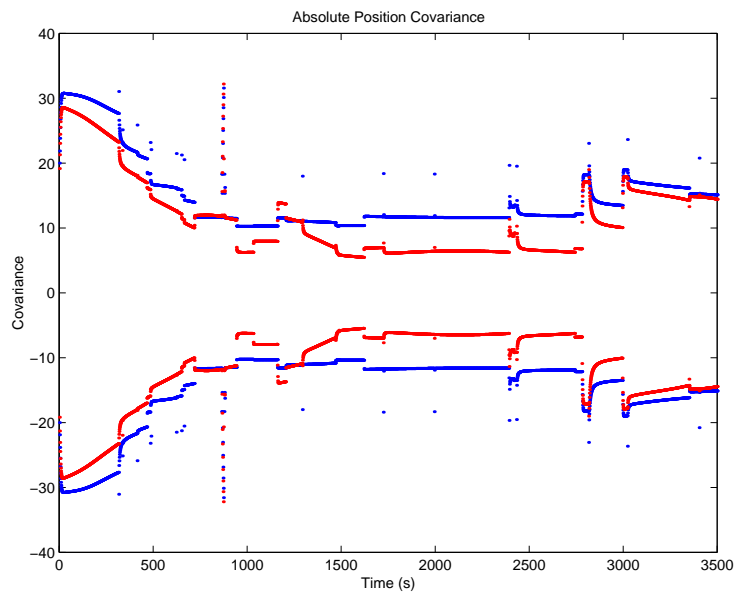


Figure C.2: Absolute Position Covariances Updates

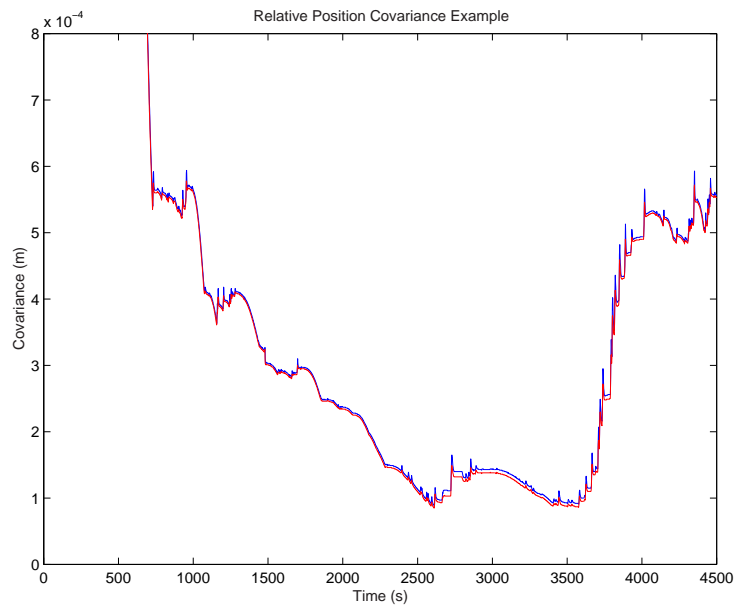


Figure C.3: Relative State Covariance Updates

Bibliography

- [1] *GPS Orion 12 Channel GPS Reference Design*, 2.0 edition, October 2001. AN4808.
- [2] Bauer, F. et al. “Enabling Spacecraft Formation Flying Through Space-born GPS and Enhanced Autonomy Technologies”. *AIAA Guidance Navigation and Control*, September 1999.
- [3] Binning, P. *Absolute and Relative Satellite to Satellite Navigation Using GPS*. PhD thesis, University of Colorado, 1997.
- [4] Brogran, W. *Modern Control Theory*, volume 3rd. Prentice Hall, 1991.
- [5] Busse, F. and J. How. “Real-Time Experimental Demonstration of Precise Decentralized Relative Navigation for Formation Flying Spacecraft”. *AIAA Guidance Navigation and Control*, August 2002.
- [6] Carpenter, J. “Decentralized Control - Theoretical Development”. *Internal Memo*.
- [7] Carpenter, J. “Partially Decentralized Control Architectures for Satellite Formations”. *Proc. of AIAA GNC Conference*, Monterey, CA, August 2002.
- [8] Carpenter, J. Decentralized control of satellite formations. *International Journal of Robust and Nonlinear Control*, (12):141–161, 2002.

- [9] Ebinuma, T. *Precision Spacecraft Rendezvous Using Global Positioning System: An Integrated Hardware Approach*. PhD thesis, The University of Texas at Austin, August 2001.
- [10] Ebinuma T., R. Bishop, and G. Lightsey. “Spacecraft Rendezvous Using GPS Relative Navigation”. AAS/AIAA Spaceflight Mechanics Meeting, February 2001 Santa Barbara, CA.
- [11] Ebinuma, T., O. Montenbruck, and G. Lightsey. “Precise Spacecraft Relative Navigation Using Kinematic Inter-Spacecraft State Estimates”. *ION GPS-2002*, September 2002.
- [12] Farrell, J. and M. Barth. *The Global Positioning System and Inertial Navigation*. McGraw-Hill, New York, 1999.
- [13] Folta, D. and A. Hawkins. “E0-1 Technology Validation Report Enhanced Formation Flying”. <http://eo1.gsfc.nasa.gov/miscPages/TechForumReports/EFF-GSFC-2.pdf>, 2001.
- [14] Gramling, C. et al. “Autonomous Relative Navigation For Formation-Flying Sattelites Using GPS”. http://geons.gsfc.nasa.gov/library_docs/ISSFDrelnavfinal.PDF.
- [15] Hoffman-Wellenhof, H., H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer-Verlag/Wien, New York, fifth edition, 2001.
- [16] Kaplan, E. *Understanding GPS: Principles and Applications*. Artech House Publishers, Norwood, MA, 1996.

- [17] Kong, E. “Minimum Energy Trajectories for Techsat-21 Earth Orbiting Clusters”. http://web.mit.edu/hilstad/www/school/16.851/FF_2-resizing.ppt, May 2001.
- [18] Lightsey, G. *Development and Flight Demonstration of a GPS Receiver for Space*. PhD thesis, Stanford University, February 1997.
- [19] Long, A. et al. “Relative Navigation Of Formation-Flying Satellites”. http://geons.gsfc.nasa.gov/library_docs/ISFF02-relnav.pdf.
- [20] Montenbruck, O. and G. Eberhard. *Satellite Orbits: Models, Mehtods, and Applications*. Springer-Verlag, Berlin, Germany, 2000.
- [21] Montenbruck, O., M. Markgraf, S. Leung, and E. Gill. “A GPS Receiver for Space Application”. *ION GPS 2001*, Salt Lake City, UT, Sep. 12-14 2001.
- [22] NASA. “Distributed Space Systems Mission List”. December 2003.
- [23] NASA. “Formation Flying: The Afternoon ‘A-Train’ Satellite Constellation”. http://eospsso.gsfc.nasa.gov/ftp_docs/A-Train_Fact_Sheet.pdf, December 2003.
- [24] Sabol C., R. Burns, and C. McLaughlin. “Satellite Formation Flying Design and Evolution”. *Spaceflight Mechanics*, v 102 pp 265-384, February 1999.
- [25] AFRL Space Vehicles Directorate. “Techsat-21 Space Missions Using Satellite Clusters”. <http://www.vs.afrl.af.mil/Factsheets/techsat21.html>, june 1999.

- [26] Spirent. *Spirent GSS Users Manual*.
- [27] Vallado, D. *Fundamentals of Astrodynamics and Applications*. Microcosm Press, El Segundo, CA, second edition, 2001.
- [28] Bamford W., G. Lightsey, and T. Ebinuma. “Navigation of Large Autonomously Controlled Formations”. *AIAA GNC Conference*, Austin, TX, August 11, 2003.
- [29] Wolfe, J., M. Abdel-Hafez, and J. Speyer. “Effective Estimation of Relative Positions in Orbit Using Differential GPS”.

Vita

William Alfred Bamford Jr. was born July 19, 1974 in San Lorenzo California to Margaret and William Bamford Sr. William graduated from San Lorenzo High School in 1992 and was awarded the position of Salutitorian. Following high school, he attended San Jose State University where he graduated Cum Laude with a B.S. in Aerospace Engineering . While at San Jose State, William worked on the Spatnik microsatellite, focusing on structure testing and assembly. Upon graduation, William enrolled at the University of Texas at Austin in the Aerospace Engineering Department. For his Master's Thesis, William worked on orbit determination of the GLONASS satellites using NASA's GIPSY/OASIS II software. William continued on for his PhD studying navigation of formations using GPS receivers as primary sensors.

Permanent address: 440 Retford Dr.
Severna Park, MD 21146
Email: `bamford@csr.utexas.edu`

This dissertation was typeset with \LaTeX^\dagger by the author.

[†] \LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's \TeX Program.